



Knowledge Advisor

- [■ Préface](#)
- [■ Nouveautés](#)
- [■ Mise en route](#)
- [■ Tâches utilisateur](#)
- [■ Description de l'atelier](#)
- [■ Langage des règles de connaissance](#)
- [■ Travaux pratiques](#)
- [■ Glossaire](#)

 P1

 P2



Préface

La conservation et la réutilisation du savoir-faire constituent l'un des enjeux majeurs de toute entreprise créant de nouveaux produits. Si toutes les entreprises produisent des données et des informations liées à leurs produits, la transformation de ces données en une base d'informations accessibles pose de réelles difficultés. Les outils utilisés dans le processus de développement des produits ne permettent généralement pas de consigner le mode et le motif des décisions prises, et ne facilitent pas davantage l'accès à ces informations. Ce savoir-faire, cette connaissance comme nous l'appellerons dans le contexte de CATIA, doit regrouper l'expérience et les informations sous une forme directement utilisable. Savoir comment regagner son domicile après le travail est une connaissance. La carte qui indique le chemin à suivre constitue les données que vous devez interpréter. Grâce à l'expérience, vous savez emprunter différents itinéraires indiqués sur la carte, en fonction de conditions variables telles que le temps ou l'heure de la journée. Vous avez utilisé la carte et d'autres informations pour créer une connaissance. De même, s'agissant du développement de produits, savoir comment un produit est censé fonctionner et comment appliquer ce savoir pour créer un produit pertinent constitue également une connaissance. Les modèles et les dessins du produit font partie des données qui le définissent mais ils n'expriment ni la méthode, ni la raison ayant amené quelqu'un à le créer de cette façon. Il est difficile d'incorporer les connaissances à une réalisation conceptuelle. Il est également difficile d'écrire ou de codifier un processus de décision de telle manière qu'il puisse être transmis sous forme de connaissance d'une personne à une autre.

Les outils le plus souvent utilisés dans le développement de produits doivent également être capables de consigner ces connaissances et permettre aux développeurs de les appliquer à la conception de nouveaux produits.

Outre une nouvelle architecture bénéficiant de plusieurs années d'évolution, CATIA Version 5 offre des fonctions très performantes qui permettent aux concepteurs de conduire les processus de création à partir de spécifications associatives et d'intégrer ces spécifications dans les procédures de développement des produits. CATIA Version 5 permet de créer des spécifications et des règles, mais aussi de les conserver et de les réutiliser.

Nouveautés

Structure du document

La documentation relative aux paramètres, aux formules et aux tables de paramétrage se trouve dans le manuel *CATIA Infrastructure - Guide de l'utilisateur*. Cette documentation décrit les tâches interactives ainsi que le langage des règles de connaissance de l'infrastructure. Un [exemple complet des principales fonctions de règles de connaissance](#) figure également dans le manuel *CATIA Infrastructure - Guide de l'utilisateur*. Dans cette exemple, vous apprendrez à utiliser simultanément certaines fonctions de règles de connaissance afin de concevoir une bouteille, améliorer sa forme ainsi que son volume.

Nouvelles Tâches

[Utilisation de règles et de vérifications dans une copie optimisée](#)

[Ajout d'un paramètre à une entité](#)

[Instanciation de relations issues d'un catalogue](#)

Nouvelle fonction du langage

[LaunchMacroFromDoc](#) : lance une macro stockée dans un document sur la base d'une règle.

Tâches améliorées

[Création d'une règle](#) : l'exemple est modifié.

[Création d'une vérification](#) : une nouvelle icône de vérification s'ajoute dans $f(x)$ en face du paramètre de résultat de la vérification.

Divers

La barre d'outils standard permet d'accéder à l'Inspecteur de Gestion de Connaissance (voir [Mode Analyse des dépendants](#) et [Mode Analyse des antécédents](#)).

Mise en route

Nous avons choisi de vous présenter les fonctions de règles de connaissance de CATIA au travers d'un exemple simple. Dans la mesure où de nombreux concepts liés aux règles de connaissance vous sont encore peu familiers à ce stade, quelques précisions seront utiles. Il suffit de suivre pas à pas les instructions décrites ci-après pour se faire une première idée des principes des fonctions de règles de connaissance de CATIA.

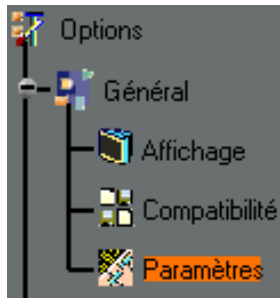
Paramètres



Cette tâche peut être réalisée par tous les utilisateurs de CATIA. Son objectif est d'expliquer brièvement la nature des paramètres et leurs principes de manipulation. Pour plus d'informations sur les tâches rattachées aux paramètres, reportez-vous au [Référentiel des tâches](#) consacré aux fonctions de règles de connaissance du programme CATIA Infrastructure.

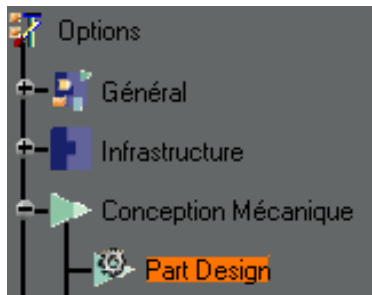


1. Cochez les paramètres ci-après :



Dans Outils ->Options ->Général ->Paramètres

- Dans l'onglet Connaissance, cochez les cases "Avec valeur" et "Avec formule".



Dans Outils ->Options ->Conception Mécanique ->Part Design

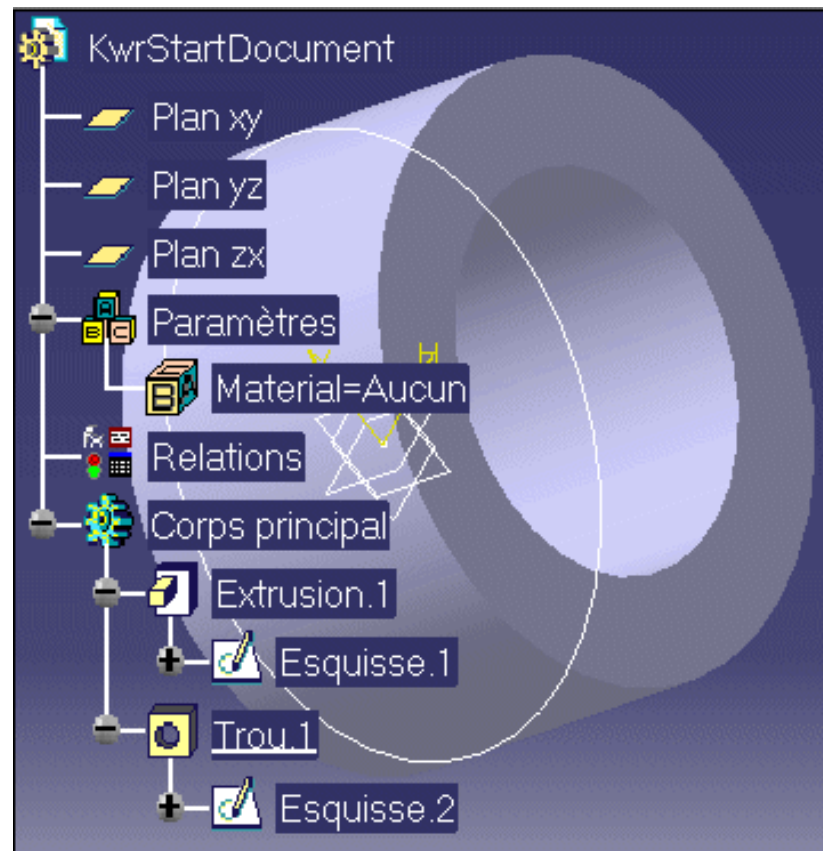
- Cochez au moins les cases "Relations" et "Paramètres. Mais il est conseillé de cocher toutes les options sous les paramètres de l'arbre des spécifications.

2. Ouvrez le document [KwrStartDocument.CATPart](#).

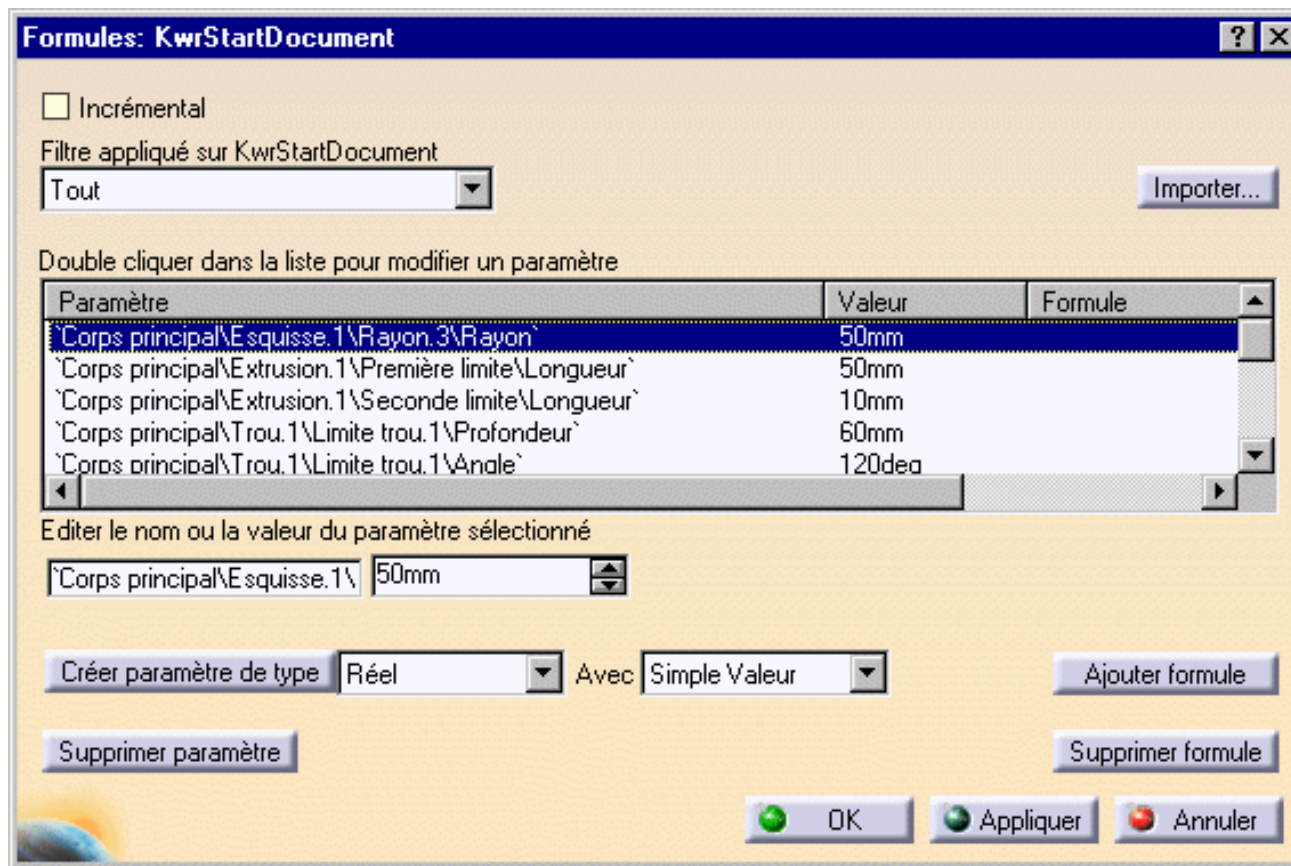
Ce document représente un cylindre creux.

Si vous développez le noeud Paramètres de l'arbre des spécifications, le paramètre Matériau est le seul à s'afficher. A ce stade du scénario, ne prêtez pas attention à ce paramètre par défaut. Nous aborderons la question ultérieurement.

Notez que le noeud Relations ne peut pas être développé dans la mesure où il n'existe pas de relation par défaut dans un document CATIA.



3. Cliquez sur l'icône . La boîte de dialogue Formules s'affiche. Vous devez désactiver la case Incrémental.



4. Sélectionnez l'option `Longueur` dans la liste `Créer paramètre de type`, puis cliquez sur `Créer paramètre de type`.
5. Dans la boîte de dialogue `Editer le nom ou la valeur du paramètre sélectionné`, remplacez la chaîne `Longueur.1` par `LongueurExtrusion`. Cliquez sur `Appliquer`. Un nouveau paramètre apparaît à présent dans la liste des paramètres de la boîte de dialogue "Formules" et dans l'arbre des spécifications. Vous venez de créer un *paramètre utilisateur*.
6. Cliquez sur `OK` dans la boîte de dialogue "Formules" pour terminer l'opération. Gardez le document ouvert et passez à la tâche suivante.




Formules

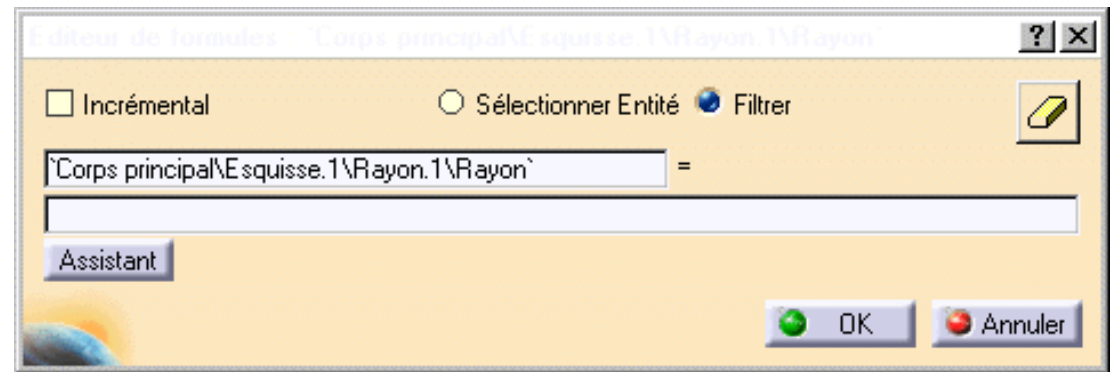


Cette tâche peut être réalisée par tous les utilisateurs de CATIA. Elle montre comment imposer une contrainte à un paramètre par le biais d'une formule. Pour plus d'informations sur les formules, reportez-vous au [Référentiel des tâches](#) consacré aux fonctions de règles de connaissance de Catia Infrastructure.



1. Cliquez sur l'icône . La boîte de dialogue "Formules" s'affiche.
2. Dans la liste des paramètres, sélectionnez l'élément `CorpsPrincipal\Esquisse.1\Rayon.1\Rayon` et cliquez sur Ajouter formule. L'éditeur de formules s'affiche.

Le champ supérieur de l'éditeur affiche le paramètre à assortir d'une contrainte. Le champ vide est pour la formule. L'icône située à droite n'est qu'une gomme.



3. A l'aide de l'assistant, entrez la relation $2 * \text{CorpsPrincipal}\backslash\text{Trou.1}\backslash\text{Diamètre}$.

Le dictionnaire est une liste complète des composants pouvant être utilisés dans une formule. Les éléments affichés peuvent être légèrement différents de ceux de la figure ci-contre.

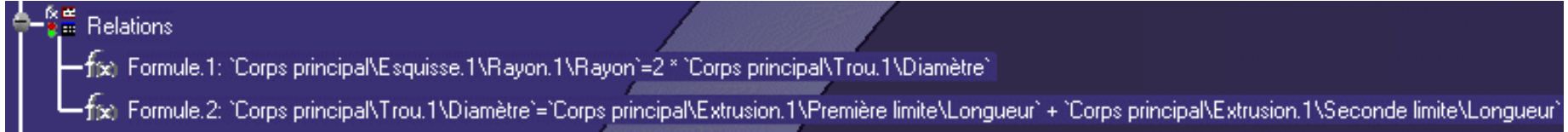


4. Une fois la relation saisie, cliquez sur OK dans l'éditeur de formules. La relation Formule.1 est ajoutée à l'arbre des spécifications. Dans la liste des paramètres de la boîte de dialogue, une formule est à présent associée au rayon de l'extrusion.
5. Dans la liste des paramètres, sélectionnez l'élément LongueurExtrusion, puis cliquez sur Ajouter formule pour créer la formule ci-dessous.

`LongueurExtrusion = CorpsPrincipal\Extrusion.1\PremièreLimite\Longueur`

+ CorpsPrincipale\Extrusion.1\SecondeLimite\Longueur

Dans la liste des paramètres, la relation Formule.2 est à présent associée au paramètre utilisateur LongueurExtrusion. Dans l'arbre des spécifications, LongueurExtrusion apparaît également avec la valeur produite par Formule.2. Voici ce que vous devez voir maintenant dans l'arbre des spécifications, sous Relations :



6. Cliquez sur OK dans la boîte de dialogue Formules pour terminer l'opération. Gardez le document ouvert et passez à la tâche suivante.




Règles



Contrairement aux fonctions liées aux paramètres et aux formules, qui sont accessibles à tous les utilisateurs de CATIA, les fonctions associées aux règles et aux vérifications requièrent le programme Knowledge Advisor. Cette tâche permet de présenter les règles de Knowledge Advisor sans entrer dans le détail. Pour plus d'informations sur les tâches rattachées aux règles, reportez-vous à la section [Tâches de gestion des règles et des vérifications](#).

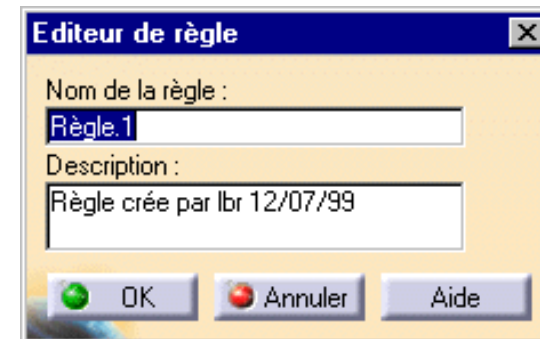


1. Sélectionnez l'élément KwrStartDocument dans l'arbre des spécifications, puis ouvrez l'atelier Knowledge Advisor (à partir du menu Démarrage->Infrastructure).

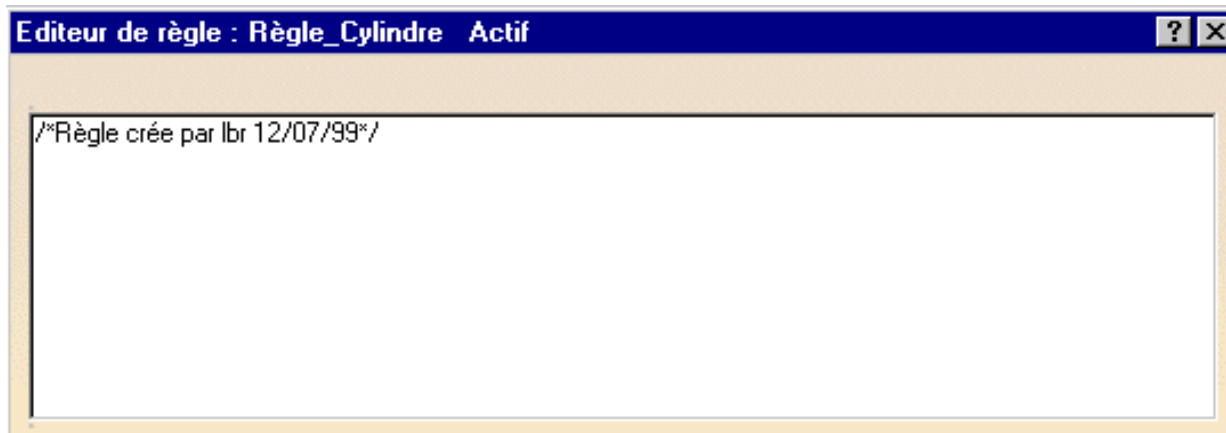
2. Cliquez sur l'icône de la règle . La boîte de dialogue suivante s'affiche :

Les deux champs affichent les valeurs par défaut pouvant être modifiées :

- Le nom de règle : Règle.i. La première règle créée dans un document est Règle.1. Ce nom s'affiche dans l'arbre des spécifications à moins que vous ayez modifié le nom par défaut à sa création.
- L'utilisateur et la date de création.



3. Remplacez la chaîne Règle.1 par RègleCylindre et cliquez sur OK. L'éditeur de règles s'affiche.



4. Tapez le code ci-dessous dans la zone d'édition des règles ou insérez-le à partir du navigateur avec un copier-coller.

```
CorpsPrincipal\Trou.1\Activité = vrai
if LongueurExtrusion <= 50mm and LongueurExtrusion > 20mm
{
CorpsPrincipal\Trou.1\Diamètre = 20mm
Message("LongueurExtrusion = # | Diamètre interne = #",
LongueurExtrusion,CorpsPrincipal\Trou.1\Diamètre)
}
else if LongueurExtrusion > 50mm and LongueurExtrusion < 100mm
{
CorpsPrincipal\Trou.1\Diamètre = 50mm
Message("LongueurExtrusion = # | Diamètre interne = #",
LongueurExtrusion,CorpsPrincipal\Trou.1\Diamètre)
}
else if LongueurExtrusion >= 100mm
{
CorpsPrincipal\Trou.1\Diamètre = 80mm
Message("LongueurExtrusion = # | Diamètre interne = #",
LongueurExtrusion,CorpsPrincipal\Trou.1\Diamètre)
}
else
{
CorpsPrincipal\Trou.1\Activité = faux
Message("LongueurExtrusion = # | Diamètre interne = #",
LongueurExtrusion,CorpsPrincipal\Trou.1\Diamètre)
}
```

5. Cliquez sur Appliquer. Une fenêtre d'information affiche la valeur du paramètre utilisateur LongueurExtrusion et celle du diamètre interne de l'extrusion. Cliquez sur OK dans la fenêtre d'information. La relation RègleCylindre est ajoutée à l'arbre des spécifications.
6. Cliquez sur OK pour clore cette partie de la transaction. Gardez le document ouvert et passez à la tâche suivante.




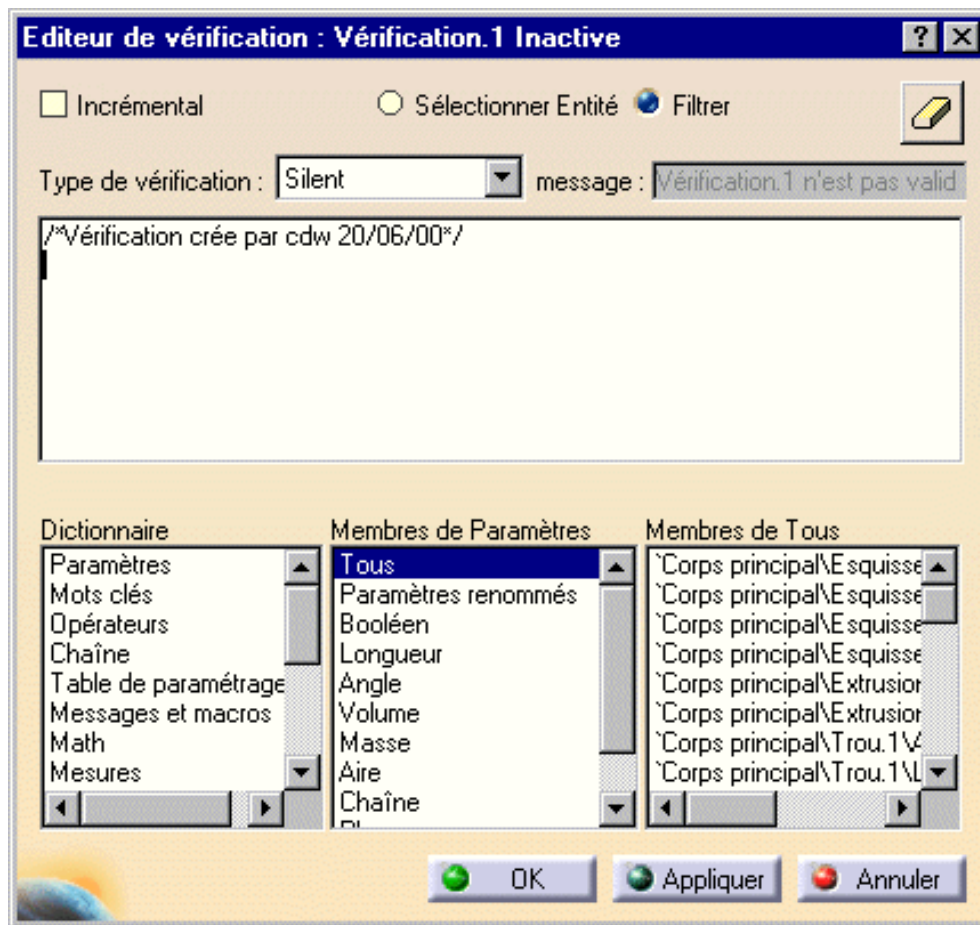
Vérifications



Une vérification est tout simplement une façon de vérifier si une instruction a abouti. Cet exemple illustre la création d'une vérification et explique ce qui se passe lorsque vous ajoutez une vérification à un document. Le programme Knowledge Advisor est nécessaire pour cette tâche. Pour plus d'informations sur les tâches rattachées aux vérifications, reportez-vous à la section [Tâches de gestion des règles et des vérifications](#).



1. Cliquez sur l'icône . La première boîte de dialogue de l'éditeur de vérifications s'affiche.
2. Remplacez le nom par défaut Vérification.1 par VérifCylindre, puis cliquez sur OK. La fenêtre de l'éditeur de vérifications s'affiche. Elle est semblable à celle de l'éditeur de règles. Vous devez désactiver la case Incrémental.



3. Sélectionnez Information dans la liste des types de vérification.
4. Entrez un texte dans la zone de message (par exemple : `Extrusion trop courte`). Ce message s'affichera chaque fois que la condition spécifiée par la vérification ne sera pas satisfaite.
5. Entrez l'instruction suivante dans la zone d'édition : `LongueurExtrusion >20mm`.
6. Cliquez sur OK pour confirmer la création de la vérification. La relation VérifCylindre est ajoutée à l'arbre des spécifications. La présence d'une icône verte dans l'arbre des spécifications indique que la vérification a réussi. Aucun message ne s'affiche dans ce cas.
7. Entrez à présent l'instruction : `LongueurExtrusion <= 20mm`. La relation RègleCylindre est appliquée à nouveau. Une fenêtre d'information affiche la nouvelle valeur du paramètre utilisateur LongueurExtrusion et celle du diamètre interne de l'extrusion. Puis, un autre message vous avertit ("Extrusion trop courte") que la vérification n'est plus concluante. L'icône de vérification devient rouge dans l'arbre des spécifications.



Tâches utilisateur

[Règles et vérifications](#)

[Inspecteur de Gestion de Connaissance](#)

[Mise à jour des mesures](#)

[Comportement](#)

[Ajout d'un paramètre à une entité](#)



Pour plus d'informations sur les paramètres et les formules, reportez-vous à la section [Utilisation des fonctions de règles de connaissance](#) (documentation du programme Infrastructure).

Reportez-vous à la section [Référentiel des tâches](#) pour connaître la liste complète des tâches liées aux règles et aux vérifications.

Règles et vérifications

[Création d'une règle](#)

[Création d'une vérification](#)

[Utilisation de règles et de vérifications dans une copie optimisée](#)

[Instanciation des relations de règles de connaissance issues d'un catalogue](#)



Pour plus d'informations sur les paramètres, les formules et les tables de paramétrage, reportez-vous à la section [Utilisation des fonctions de règles de connaissance](#) (documentation du programme Infrastructure).

Reportez-vous à la section [Référentiel des tâches](#) pour connaître la liste complète des tâches liées aux règles et aux vérifications.

Création d'une règle



Une règle est un ensemble d'instructions qui permettent de contrôler certains paramètres et événements en fonction d'un contexte. Cette tâche explique comment créer une règle qui récupère l'abscisse d'un point et, selon la valeur des coordonnées, affiche un message donné.

Cet exemple utilise deux fonctions spécifiques permettant d'extraire les coordonnées d'un point. Vous pouvez accéder à ces fonctions à partir de l'élément Mesures du dictionnaire. Pour plus d'informations, reportez-vous à la section *Langage de gestion de connaissance*. Ces fonctions sont décrites brièvement ci-après :

point.coord(NuméroCoord_i)

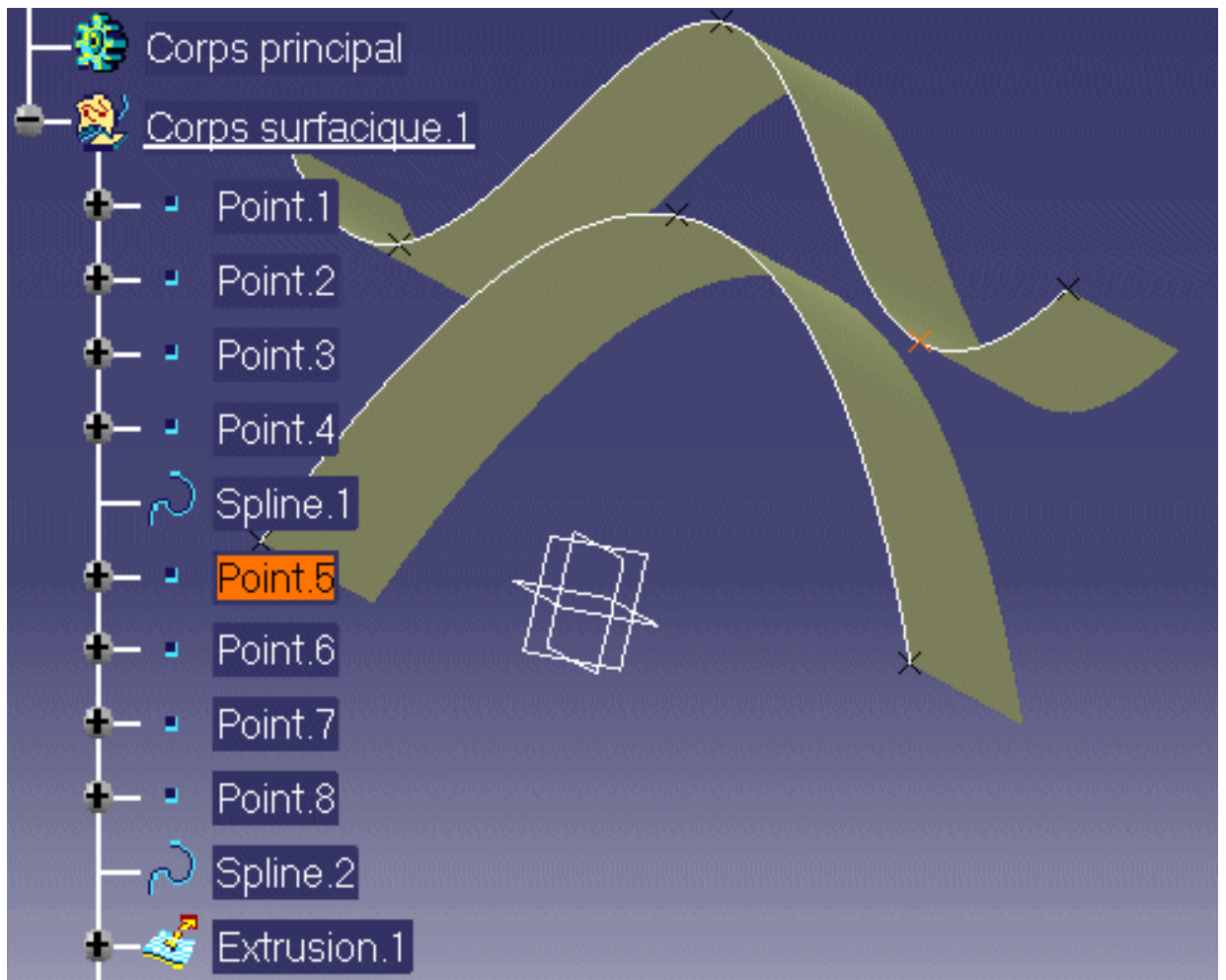
Renvoie la valeur d'une coordonnée : si *NuméroCoord* = 1, la valeur renvoyée est la coordonnée X ; si *NuméroCoord* = 2, la valeur est la coordonnée Y ; si *NuméroCoord* = 3, la valeur est la coordonnée Z.


point.coord(oRéel1, oRéel2, oRéel3)

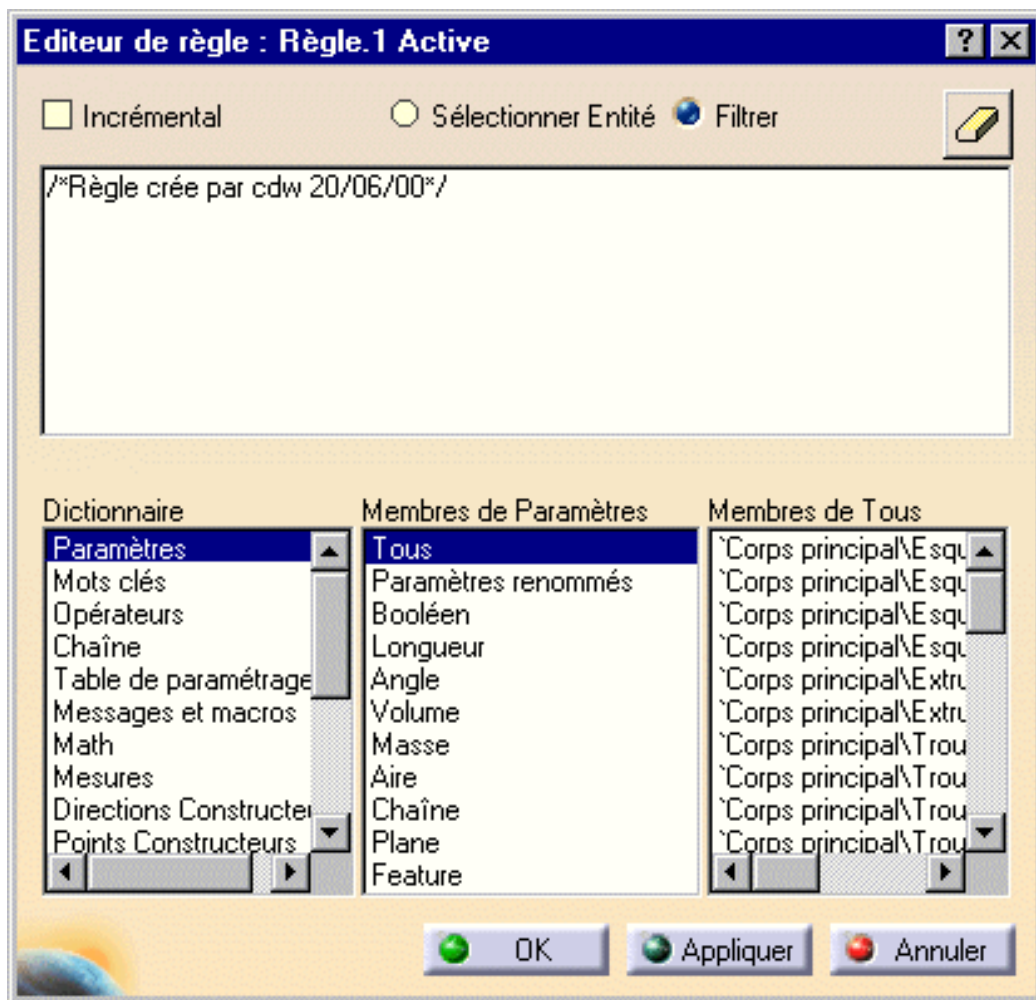
Les trois arguments de sortie sont des coordonnées de point. Vous devez spécifier les paramètres de longueur pour ces arguments.



1. Cochez la case "Chargement des bibliothèques d'extension du langage dans l'onglet Connaissance de la boîte de dialogue Outils ->Options ->Général ->Paramètres.
2. Ouvrez le document [KwrMeasure.CATPart](#). Le document a entièrement été créé avec Generative Shape Design. Les surfaces extrudées sont extrudées à partir des courbes Spline.1 et Spline.2. Le point dont les coordonnées sont à extraire et à tester est Point.5.



3. Créez trois paramètres de type Longueur : Point5X, Point5Y et Point5Z.
4. Sélectionnez l'entité racine du document et ouvrez l'atelier Knowledge Advisor.
5. Cliquez sur l'icône . Dans la première boîte de dialogue qui s'affiche, tapez le nom d'une règle (RègleMesure, par exemple). Le cas échéant, remplacez les commentaires par défaut.
6. Cliquez sur OK. L'éditeur de règles s'affiche.



7. Entrez la règle ci-dessous dans la fenêtre d'édition.

```

if Open_body.1\Point.5.coord(1) > 0mm
Message("L'abscisse de Point.5 est positive")
else
{
  Open_body.1\Point.5.coord(Point5X, Point5Y, Point5Z)
  Message("L'abscisse de Point.5 est # ", Point5X)
}

```

Dans la règle ci-dessus, vous pouvez extraire la définition du Point.5 (Open_body.1\Point.5)

- a. soit à partir d'une sélection dans l'arbre des spécifications ou dans la zone géométrique (cochez l'option Sélectionner entité),
- b. soit à partir d'une sélection dans le dictionnaire (cochez l'option Filtrer).

Cliquez sur OK. Le message, "L'abscisse de Point.5 est : 0mm", s'affiche deux fois.

8. Modifiez l'entité Point.5 (cliquez deux fois sur l'objet dans l'arbre des spécifications par exemple) et remplacez la valeur X du Point.5 par 10 mm. La règle est à mettre à jour. Pour plus d'informations sur les relations à mettre à jour, reportez-vous à la section [Mise à jour des mesures](#).

9. Ouvrez à nouveau à l'atelier Knowledge Advisor, puis cliquez sur l'icône . Un message vous informe que "l'abscisse du Point.5 est positive".



Création d'une vérification



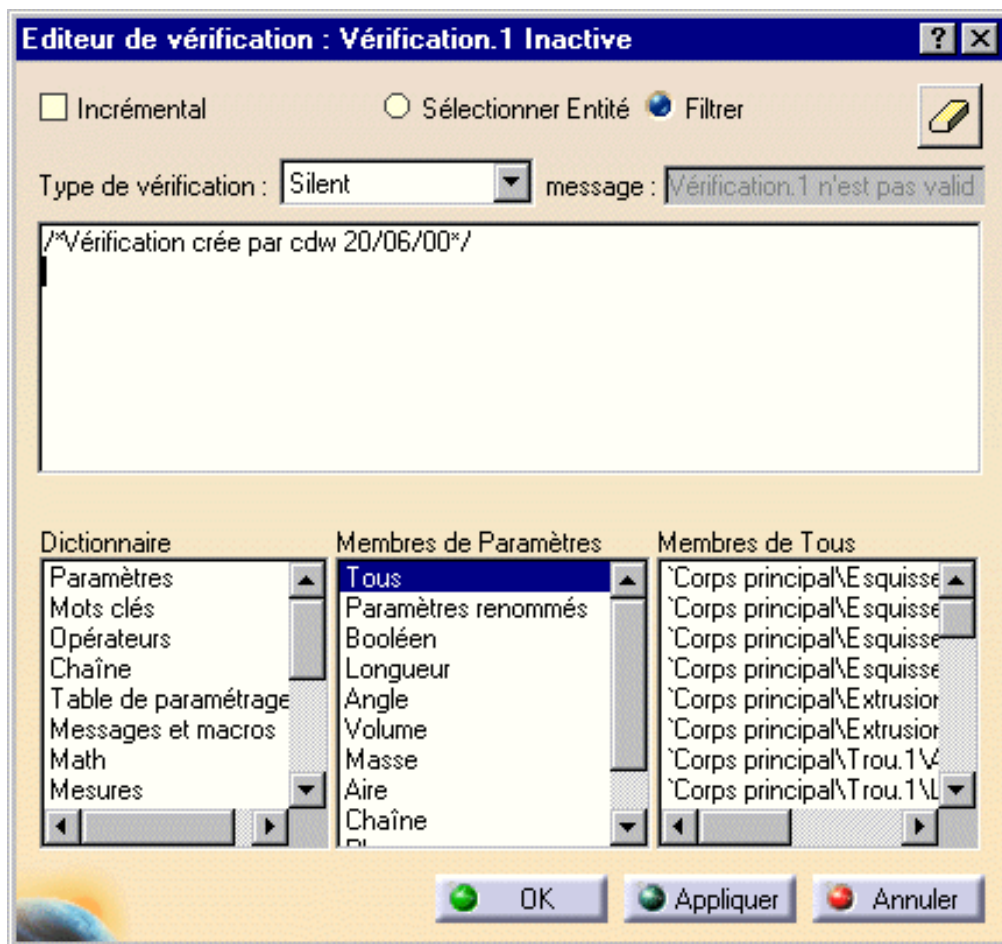
Comme une règle, une vérification est un code. Toutefois, à la différence d'une règle, une vérification est sans effet sur les valeurs des paramètres. Une fois exécutée, la vérification indique uniquement si une condition est vérifiée ou non. Il existe trois types de vérification : Silent (transparente), Information (avec informations) et Warning (avec avertissement). Selon le type de vérification choisi et le résultat de cette vérification, CATIA vous informera que :

	la vérification a réussi, ou que	la vérification a échoué.
Icône Relation dans l'arbre des spécifications		
Vérification de type Silent	pas de message affiché	pas de message affiché
Vérification de type Information	pas de message affiché	Le message spécifié lors de la création de la vérification apparaît dans une fenêtre d'information.
Vérification de type Warning	pas de message affiché	Le message spécifié lors de la création de la vérification apparaît dans une fenêtre d'avertissement.

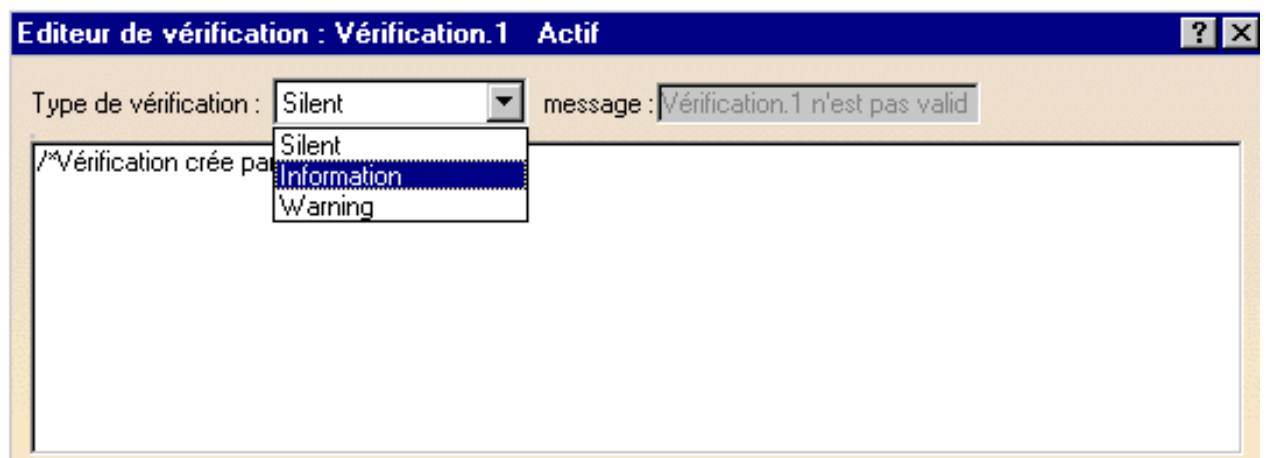
La création d'une vérification implique une part de programmation. Toutefois, la syntaxe d'un code de vérification est très simple. L'exemple suivant explique la procédure à appliquer.



1. Ouvrez le document [KwrFormula0.CATPart](#), sélectionnez l'élément racine dans l'arbre des spécifications et ouvrez l'atelier Knowledge Advisor.
2. Cliquez sur l'icône . La première boîte de dialogue de l'éditeur de vérifications s'affiche.
3. Remplacez le nom par défaut par VérifCylindre. Si nécessaire, ajoutez un commentaire dans la zone Description.



4. Cliquez sur OK. L'éditeur de vérifications s'affiche.



5. Sélectionnez un type de vérification. Entrez le message que vous voulez voir s'afficher dans la fenêtre d'information ou d'avertissement, si la vérification échoue.

6. Entrez les instructions de la vérification dans la fenêtre d'édition. Vous pouvez y insérer les instructions suivantes en faisant un copier/coller.

`Relations\Formule.1\Activité == faux`

7. Cliquez sur Appliquer pour tester la validité de la syntaxe et le résultat obtenu avec le document. La vérification VérifCylindre vient s'ajouter aux relations définies dans l'arbre des spécifications. Une icône rouge s'affiche dans l'arbre des spécifications pour indiquer que la vérification a échoué.
8. [Désactivez](#) Formula.1, l'icône de vérification devient verte dans l'arbre des spécifications.

9. Cliquez sur OK pour terminer l'opération.



Trois paramètres associés à la vérification s'affichent dans la boîte de dialogue "Formules" : l'activité, la sévérité et le résultat. Lorsque vous sélectionnez le paramètre de résultat, l'icône indiquant que la vérification a réussi ou échoué s'affiche en face du champ de valeur. Pour ouvrir l'éditeur de vérifications, double-cliquez sur cette icône.



Utilisation de règles et de vérifications dans une copie optimisée



Les règles et les vérifications ainsi que d'autres relations peuvent être appliquées à un document en les récupérant dans un autre document, à condition qu'elles aient été stockées dans une *copie optimisée*. Pour plus d'informations sur les mécanismes de copie optimisée, reportez-vous au manuel *Generative Shape Design - Guide de l'utilisateur*.




1. Ouvrez le document [KwrMeasurePCopy.CATPart](#). Le cas échéant, ouvrez l'atelier Generative Shape Design.
2. Dans la barre de menus standard, sélectionnez la commande Insérer ->Outils de réplication ->Création d'une copie optimisée. Le panneau de création de copie optimisée s'affiche.
3. Dans l'arbre des spécifications, sélectionnez les relations Règle.1 et Vérification.1. Ces deux relations sont reportées sur le panneau de création de copie optimisée. Cliquez sur OK dans le panneau de création de copie optimisée. Enregistrez et quittez le document.
4. Ouvrez le document [KwrSplineInPcopy1.CATPart](#) et l'atelier Generative Shape Design.
5. Sélectionnez la commande Insérer ->Outils de réplication ->Instanciation d'une copie optimisée dans la barre de menus standard. La boîte de dialogue "Sélection de copie optimisée" s'affiche. Sélectionnez le document contenant la copie optimisée où sont stockées les relations Règle.1 et Vérification.1, puis cliquez sur Ouvrir. La boîte de dialogue "Insertion d'objet" s'affiche.
6. Sélectionnez l'entité Spline.1 dans l'arbre des spécifications ou dans la zone géométrique. Un message généré par la vérification s'affiche, vous informant que la longueur de la spline est <100mm. Les deux relations sont reportées dans l'arbre des spécifications et l'icône de vérification est rouge. La règle n'a pas été déclenchée.
7. Ouvrez le document [KwrSplineInPcopy2.CATPart](#) et recommencez la même opération. La longueur de la spline apparaît dans une boîte d'information indiquant que la règle Règle.1 est déclenchée. Cette fois, l'icône de vérification devient verte dans l'arbre des spécifications.





Les règles et les vérifications peuvent être stockées dans des catalogues et instanciées ultérieurement dans un document. Reportez-vous à la section [Instanciation des relations de règles de connaissance issues d'un catalogue](#).




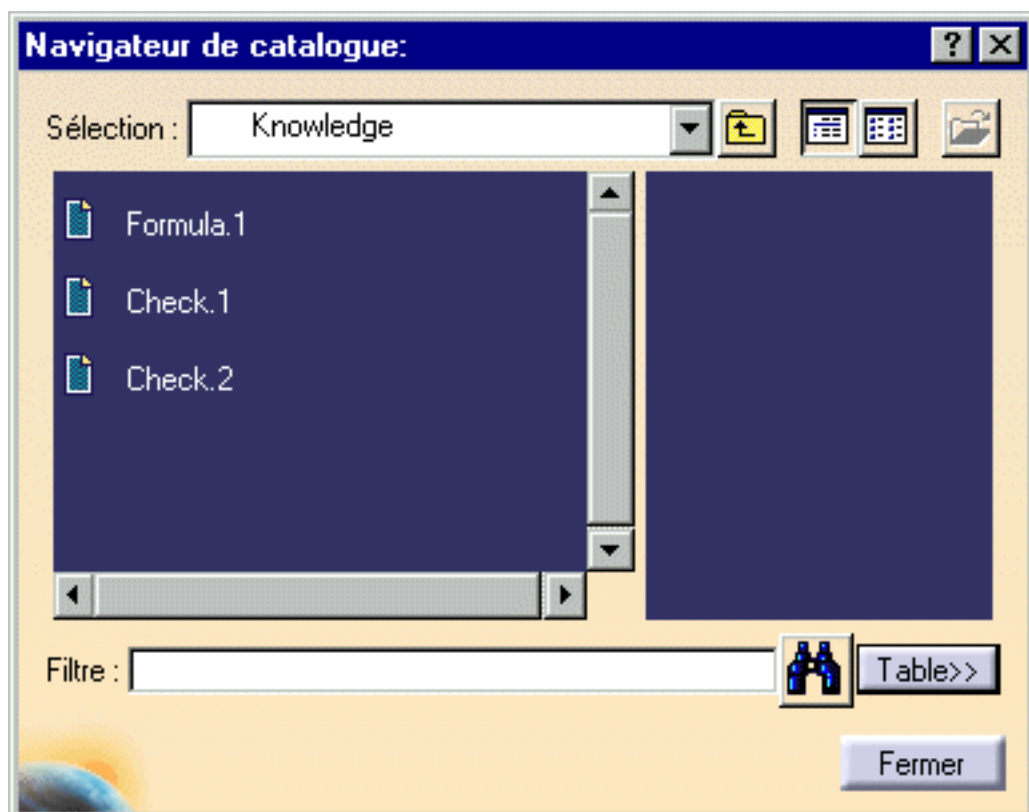
Instanciation de relations issues d'un catalogue

 Les formules, les règles et les vérifications peuvent être stockées dans un catalogue. Elles peuvent alors être réutilisées dans un document à l'aide d'un mécanisme d'instanciation. L'exemple ci-après explique comment instancier sous la forme d'un document CATPart une vérification stockée dans un catalogue. Pour plus d'informations sur les catalogues, reportez-vous au manuel *CATIA Infrastructure - Guide de l'utilisateur*.

 1. Créez une nouvelle pièce à l'aide de la commande Fichier ->Nouveau. Ensuite, créez deux paramètres de type Longueur : Rayon et Hauteur. Le cas échéant, reportez-vous à la section *Utilisation des fonctions de règles de connaissance* (Création d'un paramètre) dans le manuel *CATIA Infrastructure - Guide de l'utilisateur*.

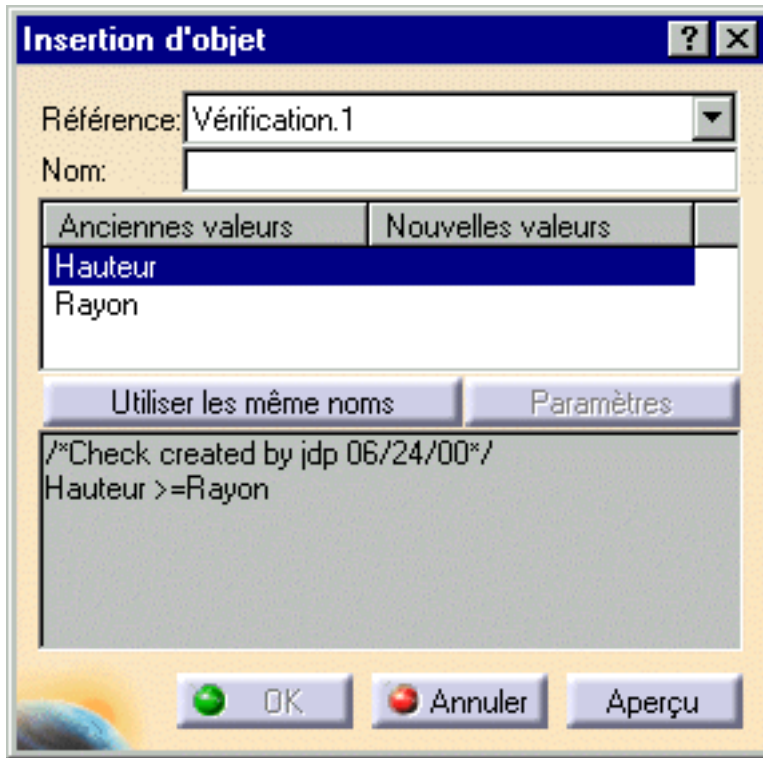
2. Dans la barre de menus standard, cliquez sur l'icône . Le catalogue de composant s'affiche.

3. Cliquez sur l'icône  pour ouvrir le catalogue CatalogFormula.catalog. Ce dernier doit être utilisé avec le document Formula_005_Start.CATPart. Le catalogue de composant ressemble à ceci :



4. Cliquez deux fois sur l'objet Vérification.1. La boîte de dialogue suivante

s'affiche.



5. Renommez la vérification Vérification.1 dans la zone Nom. Par exemple, entrez HeightCheck.
6. L'entrée Hauteur est mise en évidence. Dans l'arbre des spécifications des pièces, sélectionnez le paramètre Hauteur. L'entrée Rayon est mise en évidence. Dans l'arbre des spécifications, sélectionnez le paramètre Rayon. Cliquez sur OK. HeightCheck s'ajoute à l'arbre des spécifications et, selon les valeurs affectées aux paramètres Hauteur et Rayon, un message peut apparaître.
7. Double-cliquez sur la relation HeightCheck dans l'arbre des spécifications. La relation ci-après s'affiche dans l'éditeur de vérifications :
Hauteur >= Rayon.



Les relations d'un catalogue doivent être instanciées indépendamment les unes des autres dans un document.



Inspecteur de Gestion de Connaissance

L'Inspecteur de Gestion de Connaissance permet d'identifier et de prévisualiser les résultats de la modification des paramètres d'un document sans que ces modifications ne soient effectivement mises en oeuvre. Cette "analyse des dépendants" donne un aperçu immédiat des résultats à venir et permet d'affiner la conception des documents créés.

S'il est important de pouvoir déterminer l'effet de la modification d'un ou de plusieurs paramètres, il est également souhaitable de pouvoir déterminer comment modifier son document pour obtenir le résultat désiré. L'Inspecteur de Gestion de Connaissance répond à ce souhait en vous permettant de demander "comment" réaliser une modification donnée.

En somme, l'Inspecteur de Gestion de Connaissance est un outil qui permet d'étudier les effets et les conditions requises des modifications.

Analyse des dépendants (effets)	La fonction Analyse des dépendants permet de comprendre comment la modification d'un paramètre du document (qu'il s'agisse du matériau, de la pression ou d'une cote) modifie le fonctionnement ou la conception du produit sur lequel vous travaillez. Elle permet d'examiner l'interaction existant entre les paramètres et avec les règles qui composent les spécifications du produit. L'option Mise à jour de la géométrie permet de visualiser le résultat des modifications dans la zone géométrique.
Analyse des antécédents (dépendances)	L'option Analyse des antécédents permet de déterminer comment modifier le produit pour obtenir un résultat donné.



N'utilisez pas les fonctions  avec l'Inspecteur de Gestion de Connaissance.

[Mode Analyse des dépendants](#)

[Mode Analyse des antécédents](#)




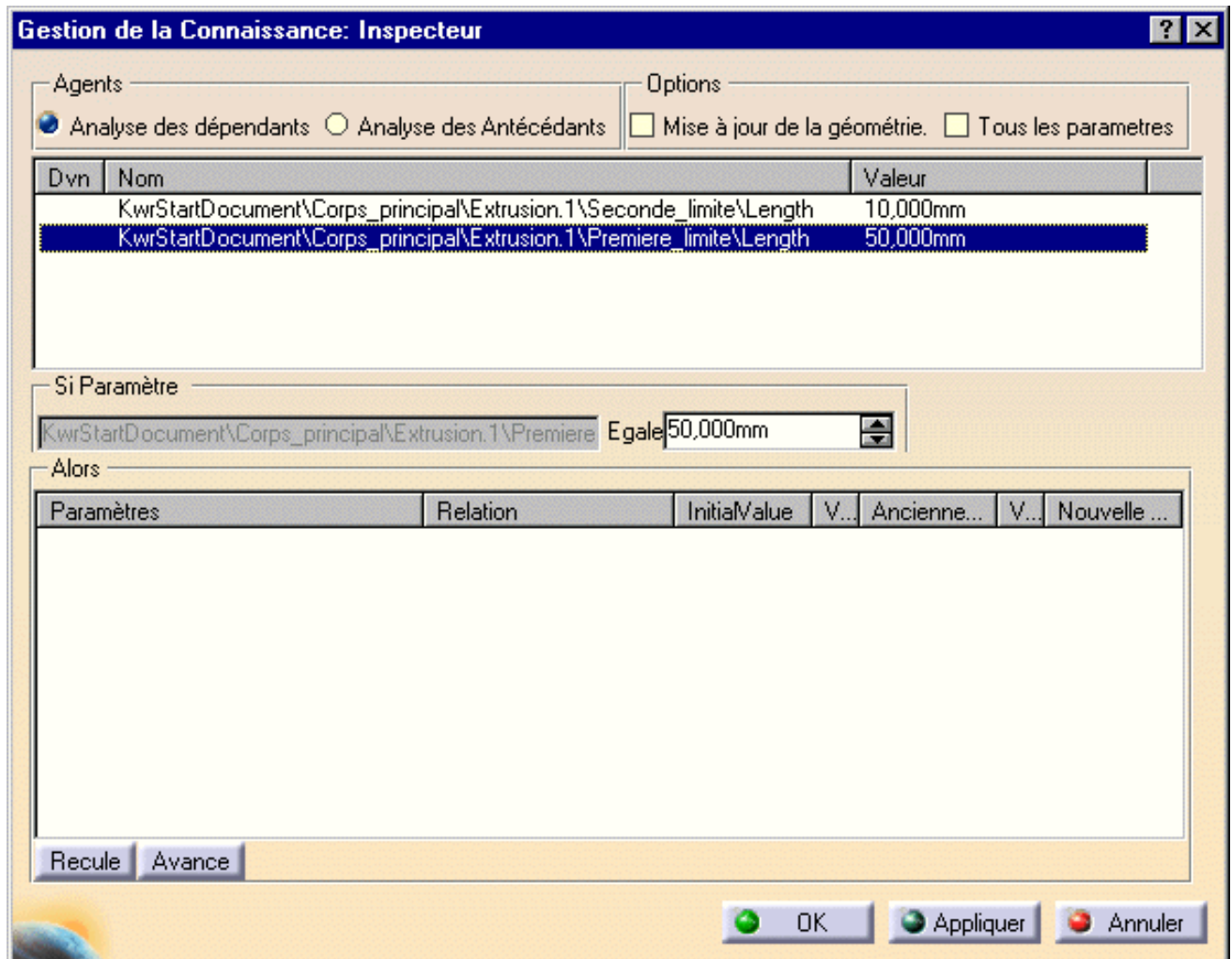
Mode Analyse des dépendants



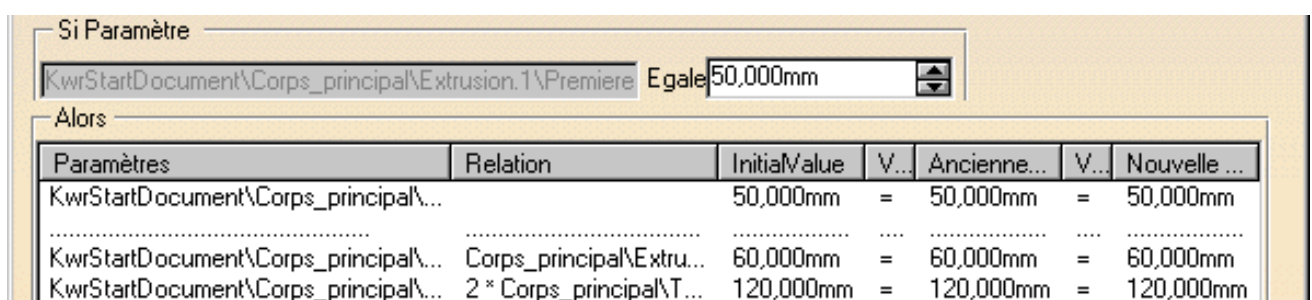
L'exemple suivant illustre les spécificités du mode Analyse des dépendants.



1. Ouvrez le document [KwrFormula1.CATPart](#) ainsi que l'atelier Knowledge Advisor.
2. Cliquez sur l'icône de l'Inspecteur de Gestion de Connaissance  ou sélectionnez-le dans la barre d'outils standard. La boîte de dialogue de l'Inspecteur de Gestion de Connaissance s'affiche. Sélectionnez l'option Analyse des dépendants.
3. Sélectionnez le paramètre Part1\CorpsPrincipal\Extrusion.1\PremièreLimite\Longueur (ne modifiez pas sa valeur dans le champ Est égal à).



4. Cliquez sur Appliquer. La liste de paramètres et de valeurs suivante s'affiche dans la zone Alors.



La première ligne décrit le dernier paramètre sélectionné. Les autres lignes décrivent les paramètres dépendants.

- Utilisez le champ "Est égal à" pour remplacer la valeur du paramètre Part1\CorpsPrincipal\Extrusion.1\PremièreLimite\Longueur par 60 mm. Cliquez sur Appliquer. Dans la zone Alors, les valeurs des paramètres se modifient de la manière suivante :

Si Paramètre

KwrStartDocument\Corps_principal\Extrusion.1\Première Limite Egale 60,000mm

Alors



Paramètres	Relation	InitialValue	V...	Ancienne...	V...	Nouvelle ...
KwrStartDocument\Corps_principal\...		50,000mm	=	50,000mm	<	60,000mm
.....
KwrStartDocument\Corps_principal\...	Corps_principal\Extru...	60,000mm	=	60,000mm	<	70,000mm
KwrStartDocument\Corps_principal\...	2 * Corps_principal\T...	120,000mm	=	120,000mm	<	140,000mm

La colonne InitialValue indique les valeurs initiales des paramètres (à l'ouverture de l'Inspecteur de Gestion de Connaissance). La colonne Ancienne valeur contient les valeurs des paramètres issues de l'opération Analyse des dépendants précédente. Les colonnes Var (variations) contiennent les opérateurs de comparaison utilisés entre les valeurs des colonnes adjacentes.

- Sélectionnez l'option Mise à jour géométrique pour afficher les modifications issues de l'opération Analyse des dépendants dans la zone géométrique. Cliquez sur Appliquer pour mettre à jour le document dans la zone géométrique.
- Cliquez sur OK pour appliquer au document les valeurs produites par l'opération Analyse des dépendants courante. Si vous ne voulez pas appliquer ces valeurs, cliquez sur Annuler.



Remarques :

- Cliquez sur les boutons   pour insérer dans la zone Alors les valeurs précédentes ou suivantes dans l'historique des opérations Analyse des dépendants.
- Pour afficher tous les paramètres du document, sélectionnez l'option Tous les paramètres. La lettre f, dans la colonne Dvn, indique que le paramètre est assorti d'une formule de contrainte.
- Si l'option Tous les paramètres a été cochée et que vous sélectionnez un paramètre dans la zone Alors, le paramètre choisi est mis en évidence dans la liste placée au dessus.



La modification d'une valeur de paramètre n'entraîne pas la mise à jour automatique des paramètres dépendants en cas d'opération Analyse des dépendants. Par exemple, si un paramètre est assorti d'une formule de contrainte, telle que :

if Paramètre1 > A then Paramètre2 = B

remplacer la valeur de Paramètre1 par une valeur supérieure à A ne modifiera pas la valeur de Paramètre2 si cette valeur était B, initialement.




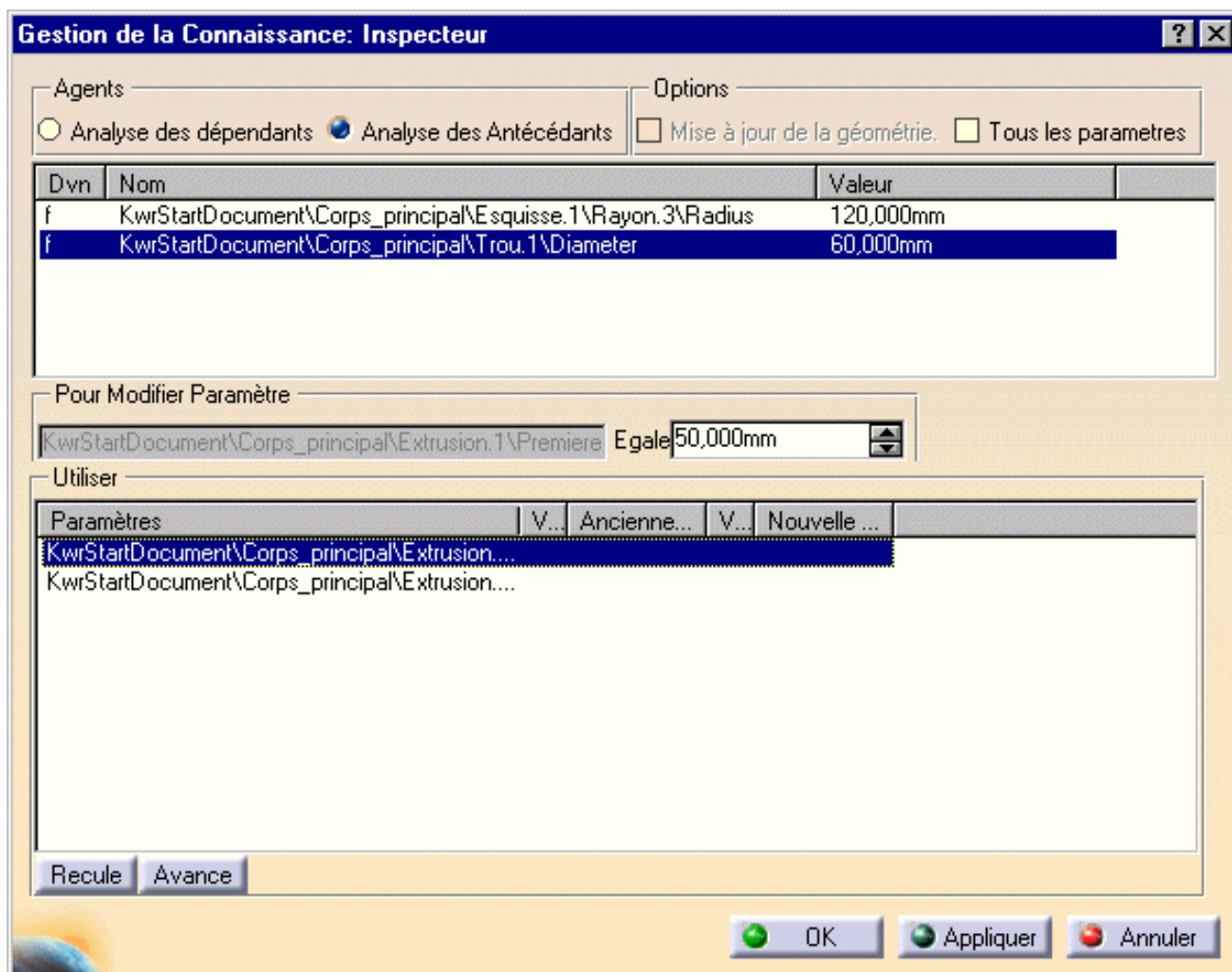
Mode Analyse des antécédents



Cette section fournit des informations sur l'utilisation du mode Analyse des antécédents.



1. Ouvrez le document [KwrFormula1.CATPart](#) ainsi que l'atelier Knowledge Advisor.
2. Cliquez sur l'icône de l'Inspecteur de Gestion de Connaissance  ou sélectionnez-le dans la barre d'outils standard. La boîte de dialogue Inspecteur de Gestion de Connaissance s'affiche. Cochez l'option Analyse des antécédents. Par défaut, seuls les paramètres assortis d'une formule de contrainte sont affichés.
3. Le cas échéant, cochez l'option Tous les paramètres pour afficher tous les paramètres du document.
4. Sélectionnez le paramètre Part1\CorpsPrincipal\Trou.1\Diamètre (imaginons que vous souhaitez modifier ce paramètre).
5. Cliquez sur Appliquer ou sur ENTREE. La liste des paramètres à modifier pour changer la valeur du paramètre Trou.1\Diamètre s'affiche dans la zone Utiliser.
6. Sélectionnez le paramètre Extrusion.1\PremièreLimite\Longueur.



7. Sélectionnez l'option Analyse des dépendants.
8. Modifiez le paramètre PremièreLimite\Longueur en mode Analyse des dépendants.
9. Cliquez sur OK pour appliquer au document la modification du paramètre.




Remarques :

- Pour afficher tous les paramètres du document, sélectionnez l'option Tous les paramètres. La lettre f, dans la colonne Dvn, indique que le paramètre est assorti d'une formule de contrainte.
- Si l'option Tous les paramètres a été cochée et que vous sélectionnez un paramètre dans la zone Alors, le paramètre choisi est mis en évidence dans la liste placée au dessus.



Mise à jour de relations utilisant des mesures



Une relation utilisant des mesures doit être mise à jour lorsque le symbole  s'affiche en face de cette relation dans l'arbre des spécifications.

Exemple


MeasureRule requiert une mise à jour.



MeasureRule ne doit pas être mise à jour.



Pour mettre à jour les règles, vous devez :

- cliquer sur l'icône  (pour cela, vous devez être dans l'atelier Knowledge Advisor)
- ou sélectionner la commande Mise à jour des mesures dans le menu contextuel du noeud Relations. Vous pouvez procéder ainsi dans tous les ateliers.

Toutes les relations du document sont alors mises à jour.



Comportement

Un comportement se caractérise par une série d'instructions VB Script qui s'exécutent en réponse à un événement donné. Pour plus d'informations sur la liste des fonctions pouvant être utilisées dans un comportement, reportez-vous à la section [Référentiel](#).

[Création d'un comportement](#)

[Ecriture d'un comportement](#)

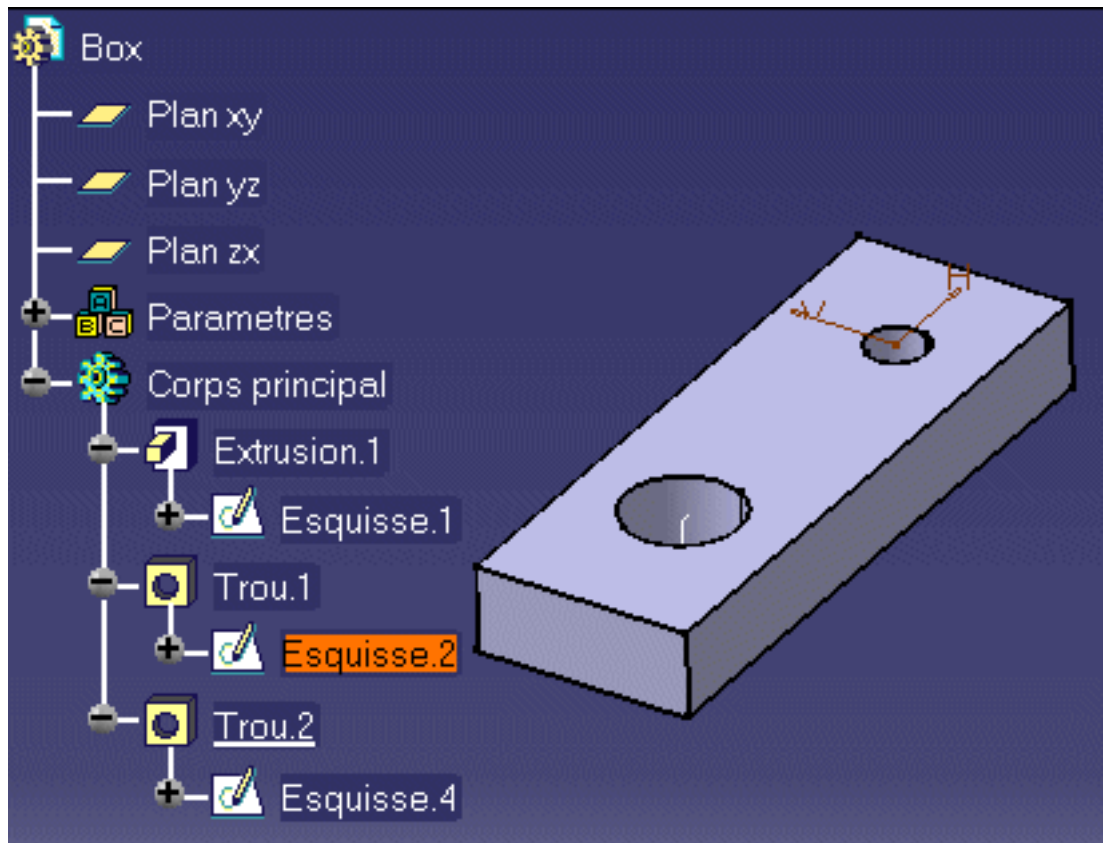
Création d'un comportement




Un comportement se caractérise par une série d'instructions VB Script associées à une entité. Ces instructions s'exécutent lorsque l'entité concernée répond à un événement donné. Actuellement, un comportement ne peut être associé qu'aux entités manipulées par glisser/déposer. Dans cette tâche, vous apprendrez à créer un comportement.



1. Créez une boîte ou une extrusion semblable à celle présentée ci-dessous.



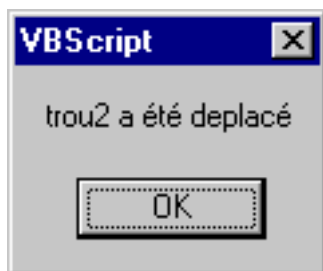
2. Sélectionnez l'élément Boîte (ou l'élément racine de votre document) dans l'arbre des spécifications.
3. Dans le menu Démarrer, sélectionnez Infrastructure-> Atelier Knowledge Advisor.
4. Sélectionnez l'entité Trou.2, soit dans la zone géométrique, soit à partir de l'arbre des spécifications.
5. Cliquez sur l'icône . La boîte de dialogue des comportements apparaît. Dans cette édition de CATIA, la liste des événements pouvant être gérés se limite à l'événement glisser/déposer. Cet événement apparaît dans le champ Lorsque de la boîte de dialogue des comportements.
6. Entrez un nom de comportement dans le champ Nom de la boîte de

dialogue des comportements.

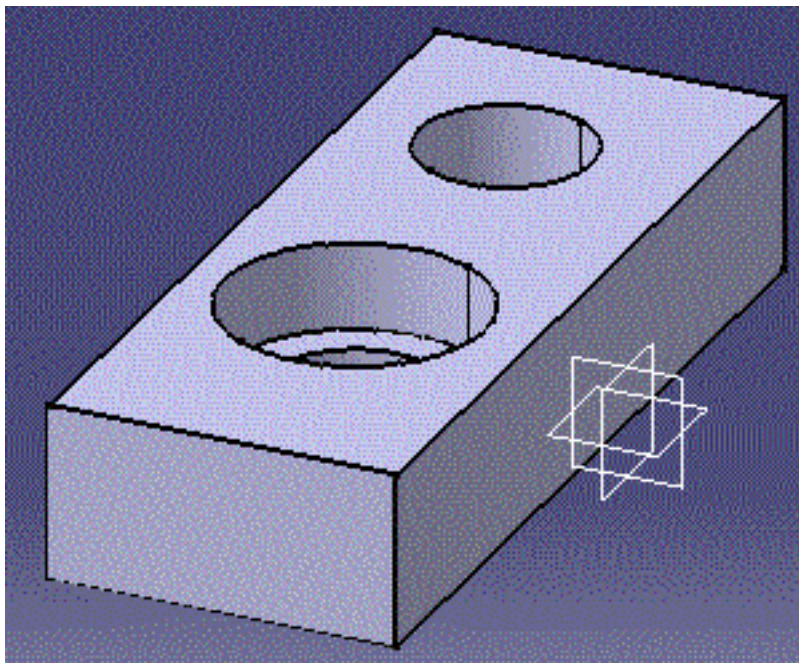
7. A partir du navigateur, insérez le code ci-dessous dans la zone d'édition de l'éditeur de comportements avec un copier-coller.

```
MsgBox("Trou.2 a été déplacé")  
MyEvent.GetDroppedFeature.Type = 2
```

8. Cliquez sur OK. Un comportement s'ajoute à l'arbre des spécifications.
9. Revenez à l'atelier de conception des pièces (double-cliquez sur le document dans la zone géométrique par exemple).
10. Rapprochez l'entité Trou2 de Trou1 par un glisser-déposer. Le message ci-dessous apparaît :



Voici ce que vous pouvez voir à l'écran. L'entité Trou2 a été convertie en un trou de type 2 (un trou alésé).



Cet exemple illustre de manière simple comment créer un comportement et comment celui-ci agit sur une entité. Pour créer une application de comportement complète, vous devez maîtriser le langage VB Script de même que les objets d'automatisation utilisables par ce type de script.





Ecriture d'un comportement




Il n'est pas toujours possible d'accéder à une entité ou de la modifier par le biais d'un comportement. Vous pouvez uniquement manipuler les objets placés sous l'objet Forme dans la hiérarchie des objets d'automatisation. Voici un exemple illustrant comment manipuler les attributs des éléments.

L'entité sur laquelle s'opère le glisser-déposer est un trou simple. Lorsque le comportement s'exécute :

1. les propriétés suivantes du trou s'affichent dans une fenêtre VB :
 - le type d'entité ;
 - le nom de l'entité ;
 - le diamètre du trou.
2. Un message demande à l'utilisateur :
 - soit de cliquer sur OK pour modifier les propriétés du trou une fois celui-ci déplacé :
Les nouvelles valeurs des propriétés sont les suivantes :
 - trou (type = 2) ;
 - diamètre = 10,0 ;
 - diamètre de la tête = 20,0.
 - soit de cliquer sur Annuler et de déplacer le trou sans modifier ses propriétés.



1. Créez une extrusion avec au moins un trou simple à modifier par glisser-déposer.
2. Sélectionnez l'entité racine et ouvrez l'atelier Knowledge Advisor.
3. Sélectionnez un trou quelconque dans la zone géométrique ou dans l'arbre des spécifications.
4. Cliquez sur l'icône . L'éditeur de comportements s'affiche. Tapez le nom du comportement (ne modifiez pas l'événement CATDropFeatureEvent dans la mesure où il est affiché par défaut dans le champ Lorsque).
5. A partir du navigateur, insérez le code ci-dessous dans l'éditeur de comportements avec un copier-coller.

```
Dim FeatureType, FeatureName, HoleType
FeatureType = TypeName(MyEvent.GetDroppedFeature)
FeatureName = MyEvent.GetDroppedFeature.Name
HoleType = MyEvent.GetDroppedFeature.Type
Str1="Entité_à_déplacer"
Str2=Str1 & vbCrLf & "est une entité de type "
& FeatureType & " "
Str3=Str2 & vbCrLf & "Son nom est '" &
FeatureName & "'"
Str4 = Str3 & vbCrLf & "Son diamètre initial est "_
& MyEvent.GetDroppedFeature.Diameter.Value & vbCrLf
Str5=Str4 & vbCrLf &_
"Il sera transformé en trou alésé si vous ne cliquez pas sur
Annuler."
Dim MyBox
MyBox = MsgBox (Str5,1)
if MyBox = 1 then
MyEvent.GetDroppedFeature.Type = 2 MyEvent.GetDroppedFeature.HeadDiameter.Value =
20.0
MyEvent.GetDroppedFeature.Diameter.Value = 10.0
End If
```


6. Cliquez sur OK. Le comportement DroppedHole s'ajoute à l'entité Trou dans l'arbre des spécifications.
7. Revenez à l'atelier de conception des pièces (double-cliquez sur le document par exemple).
8. Faites un glisser-déposer sur l'objet Trou. Les valeurs des attributs de l'objet Trou s'affichent dans une boîte de dialogue Visual Basic.
9. Cliquez sur OK pour convertir le trou déplacé en trou alésé, ou cliquez sur Annuler pour le déplacer sans modifier ses propriétés.





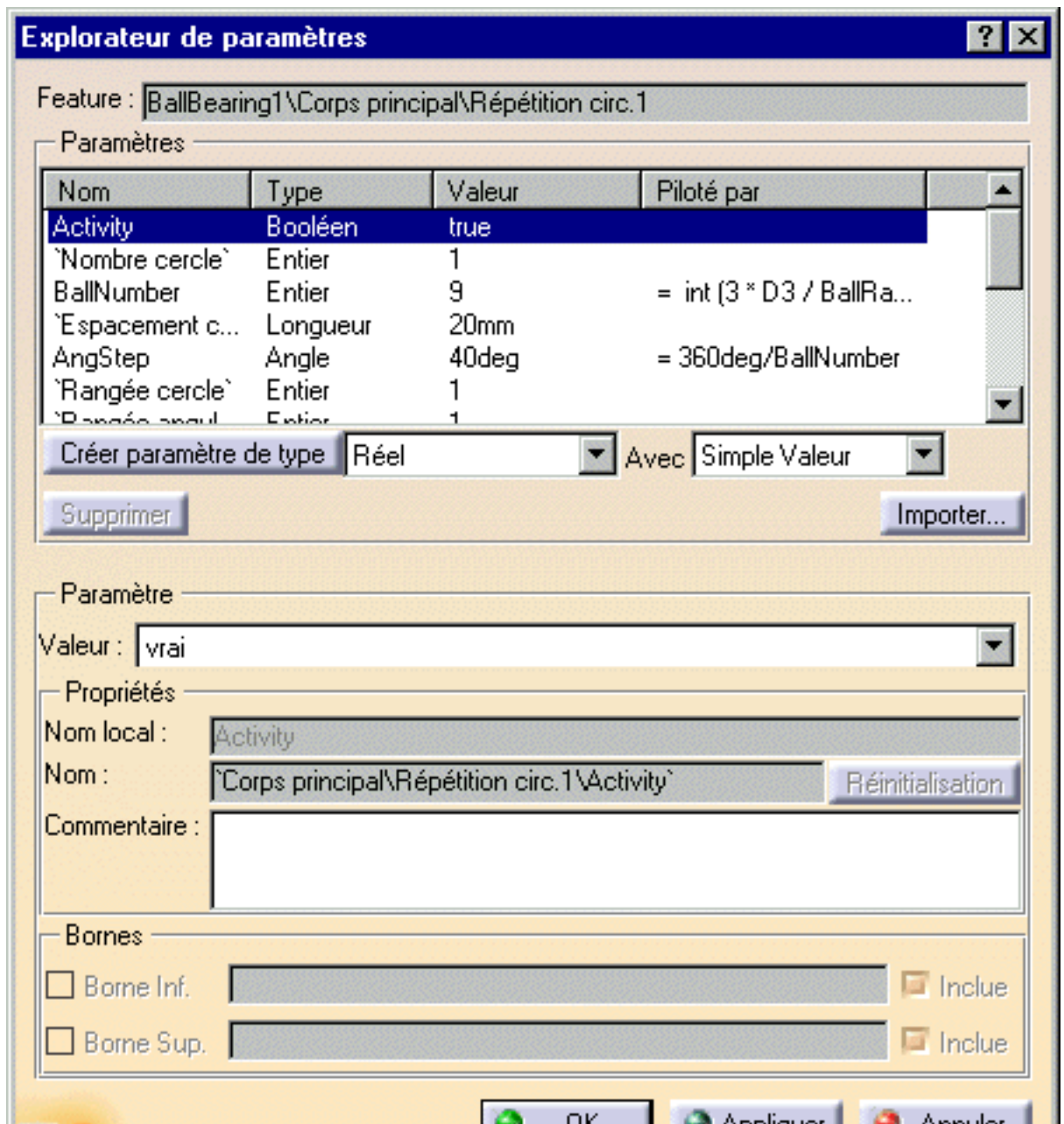
La propriété Nom renvoie le nom de l'entité une fois celle-ci déplacée. Ce nom, différent du nom d'origine, est celui que CATIA affecte par défaut à l'élément déplacé.



Ajout d'un paramètre à une entité

 Dans cette tâche, vous allez apprendre à ajouter deux paramètres à une entité de modèle circulaire. L'un de ces paramètres est une chaîne à plusieurs valeurs et l'autre une masse avec des limites supérieure et inférieure.

- 
1. Ouvrez le document [KwrBallBearing1.CATPart](#).
 2. Dans l'arbre des spécifications, sélectionnez l'entité racine, puis la commande Démarrer ->Infrastructure -> Knowledge Advisor pour ouvrir l'atelier Knowledge Advisor.
 3. Dans l'arbre des spécifications, sélectionnez CircPattern.1.
 4. Cliquez sur l'icône , la boîte de dialogue ci-après s'affiche :



Explorateur de paramètres

Feature : BallBearing1\Corps principal\Répétition circ.1

Paramètres

Nom	Type	Valeur	Piloté par
Activity	Booléen	true	
`Nombre cercle`	Entier	1	
BallNumber	Entier	9	= int (3 * D3 / BallRa...
`Espacement c...	Longueur	20mm	
AngStep	Angle	40deg	= 360deg/BallNumber
`Rangée cercle`	Entier	1	
`Rangée seul	Entier	1	

Créer paramètre de type Réel Avec Simple Valeur

Supprimer Importer...

Paramètre

Valeur : vrai

Propriétés

Nom local : Activity

Nom : `Corps principal\Répétition circ.1\Activity` Réinitialisation

Commentaire :

Bornes

Borne Inf. Inclue

Borne Sup. Inclue

OK Appliquer Annuler

5. Dans la liste Créer paramètre de type, sélectionnez le type Chaîne, puis, dans le champ situé en face ("Par"), sélectionnez l'élément Valeurs multiples. Cliquez sur Créer paramètre de type.
6. Dans la boîte de dialogue "Liste de valeurs" :
 - a)entrez la chaîne Type1, puis appuyez sur Entrée.
 - b)entrez la chaîne Type2, puis appuyez sur Entrée.
 - c)cliquez sur OK pour revenir à la boîte de dialogue "Explorateur de paramètres".
7. Le cas échéant, renommez le paramètre créé dans le champ Nom local et ajoutez un commentaire.
8. Dans la liste Créer paramètre de type, sélectionnez le type Masse, puis, dans le champ situé en face ("Par"), sélectionnez l'élément Simple valeur. Cliquez sur Créer paramètre de type. MASSE.1 s'affiche par défaut dans le champ Propriétés et la valeur par défaut 0kg est affectée au paramètre créé.
9. Modifiez ces valeurs comme indiqué dans la figure ci-dessous :

Paramètre

Valeur : 0,01kg

Propriétés

Nom local : MASS.1

Nom : Corps principal\Répétition circ.1\Masse.1 Réinitialisation

Commentaire :

Bornes

Borne Inf. 0kg Inclue

Borne Sup. 0,02kg Inclue

10. Cliquez sur OK. Les deux paramètres s'affichent dans l'arbre des spécifications, juste en dessous de l'entité CircPatter.1.



Référentiel des tâches

Tâche de gestion des paramètres

Ajout d'un paramètre à une entité [Exemple](#)



Sélectionnez une entité, puis utilisez le bouton Créer paramètre de type pour indiquer le paramètre à ajouter à l'entité sélectionnée.

Tâches de gestion des règles

Création d'une règle [Exemple](#)



Entrez le nom de la règle et des commentaires. Cliquez sur OK pour afficher l'éditeur de règles, puis écrivez la règle.

Entrée d'une définition de paramètre dans une règle



Pour entrer la définition d'un paramètre dans une règle, vous pouvez :

1. cocher la case Sélectionner entité dans l'éditeur de règles, puis sélectionner l'entité dans l'arbre des spécifications ou dans la zone géométrique ;
2. cliquer sur la cote affichée dans la zone géométrique ;
3. utiliser le dictionnaire.

Edition et modification d'une règle



Dans l'arbre des spécifications, cliquez deux fois sur la règle à éditer. L'éditeur de règles apparaît.

ou

Dans l'arbre des spécifications, avec le bouton droit, cliquez sur la règle à éditer, puis sélectionnez la commande *Objet Règle* ->Edition dans le menu contextuel.

Activation et désactivation d'une règle

Dans l'arbre des spécifications, avec le bouton droit, cliquez sur la règle à activer ou à désactiver, puis sélectionnez la commande *Objet Règle* ->Activer/Désactiver dans le menu contextuel.

ou



Dans la liste des paramètres de la boîte de dialogue "Formules", sélectionnez le paramètre de règle/activité et modifiez sa valeur dans le champ Editer le nom ou la valeur du paramètre sélectionné.

Suppression d'une règle

Dans l'arbre des spécifications, cliquez avec le bouton droit sur la règle à supprimer, puis sélectionnez la commande Supprimer dans le menu contextuel.

Tâches de gestion des vérifications

Création d'une vérification [Exemple](#)



Entrez le nom de la vérification et des commentaires. Entrez ensuite la vérification dans l'éditeur de vérifications. Pour spécifier un paramètre à l'intérieur d'une vérification, vous pouvez au choix :

1. cliquer sur le paramètre dans l'arbre des spécifications ;
2. cliquer sur la cote affichée dans la zone géométrique ;
3. utiliser le dictionnaire.

Entrée d'une définition de paramètre dans une vérification



Pour entrer la définition d'un paramètre dans une vérification, vous pouvez au choix :

1. cocher la case Sélectionner entité dans l'éditeur de vérifications, puis sélectionner l'entité dans l'arbre des spécifications ou dans la zone géométrique ;
2. cliquer sur la cote affichée dans la zone géométrique ;
3. utiliser le dictionnaire.

Edition et modification d'une vérification



Dans l'arbre des spécifications, double-cliquez sur la vérification à éditer. L'éditeur de vérifications apparaît.

ou

Dans l'arbre des spécifications, avec le bouton droit, cliquez sur la vérification à éditer, puis sélectionnez la commande *Objet Vérification* ->Edition dans le menu contextuel.

ou



Dans la liste des paramètres, sélectionnez le paramètre *Relations.il/Vérification/Résultat*. La valeur du résultat de la vérification est grisée dans la boîte de dialogue Editer le nom ou la valeur du paramètre sélectionné. Cliquez sur le bouton se trouvant en face du champ de valeur pour afficher l'éditeur de vérifications.

Activation et désactivation d'une vérification

Dans l'arbre des spécifications, avec le bouton droit, cliquez sur la vérification à activer ou à désactiver, puis sélectionnez la commande *Objet Vérification* ->Activer/Désactiver dans le menu contextuel.

ou



Dans la liste des paramètres de la boîte de dialogue "Formules", sélectionnez le paramètre de vérification/activité et modifiez sa valeur dans le champ Editer le nom ou la valeur du paramètre sélectionné.

Suppression d'une vérification

Dans l'arbre des spécifications, avec le bouton droit, cliquez sur la vérification à supprimer, puis sélectionnez la commande Supprimer dans le menu contextuel.

Tâches de l'Inspecteur de Gestion de Connaissance

Accès au mode "Analyse des dépendants" (effets) [Exemple](#)



1. Cochez l'option Analyse des dépendants, puis sélectionnez le paramètre dont les effets sont à analyser.
2. Cliquez sur Appliquer. La liste des paramètres affectés par la modification s'affiche dans la zone Alors.
3. Pour modifier la valeur du paramètre sélectionné, utilisez le champ Est égal à.
4. Pour afficher les valeurs des paramètres dépendants, cliquez sur Appliquer.

Accès au mode "Analyse des antécédents" (dépendances) [Exemple](#)



1. Cochez l'option Analyse des antécédents, puis sélectionnez le paramètre dont les dépendances sont à étudier.
2. Cliquez sur Appliquer ou sur ENTREE. La liste des paramètres intervenant dans la modification s'affiche dans la zone Alors.

Tâches de gestion des comportements

Création d'un comportement [Exemple](#)



Cliquez sur l'icône des comportements et tapez votre script dans l'éditeur qui s'affiche. Le script de ce comportement doit être écrit dans le langage Visual Basic.

Programmation d'un comportement [Exemple](#)



Objets et propriétés :

Vous pouvez uniquement manipuler les objets placés sous l'objet *Forme* dans la hiérarchie des objets d'automatisation. Les méthodes de *l'objet MyEvent* :

`GetDroppedFeature`

 Renvoie l'entité copiée.

`GetTargetOfType`

 Renvoie l'entité sur laquelle une autre entité a été déposée. Indiquez un type d'entité comme argument.

`GetX, GetY, GetZ`

 Renvoie les coordonnées de l'emplacement de la copie.

`HasCoordinates`

 Détermine si la copie est associée à des coordonnées.

Description de l'atelier

Barre de menus de Knowledge Advisor

La barre de menus de l'atelier Knowledge Advisor est la barre de menus standard. De plus, la commande Outils permet d'accéder à l'outil *Inspecteur de Gestion de Connaissance*.

Barre d'outils de Knowledge Advisor

La figure ci-dessous décrit la barre d'outils de Knowledge Advisor. Cliquez sur une icône pour afficher la documentation relative à la tâche associée.



Ces icônes sont sommairement décrites ci-après.



L'icône **Editeur de règles** permet d'accéder à l'éditeur de règles. Cliquez sur cette icône pour créer une règle, écrire son code, valider sa syntaxe et l'appliquer à votre document.



L'icône **Editeur de vérification** permet d'accéder à l'éditeur de vérifications. Cliquez sur cette icône pour créer une vérification, écrire son code, valider sa syntaxe et l'appliquer à un document.



L'**Inspecteur de Gestion de Connaissance** permet d'identifier et de prévisualiser les résultats de la modification des paramètres d'un document sans que ces modifications ne soient effectivement mises en oeuvre.



L'icône **Comportement** permet de créer un script capable de modifier certains attributs des composants lorsqu'un événement survient. Cliquez sur cette icône pour ouvrir l'éditeur de comportements. Un comportement est un script Visual Basic.



L'icône **Mise à jour des mesures** permet de mettre à jour des relations utilisant des mesures.



L'icône **Explorateur de paramètres** permet d'ajouter de nouveaux paramètres à une entité.

Langage des règles de connaissance

Cette section concerne les utilisateurs de CATIA souhaitant utiliser des fonctions avancées des produits CATIA Infrastructure et CATIA Knowledge Advisor.

[Ecriture de règles et de vérifications](#)

[Ecriture d'un comportement](#)



Pour plus d'informations sur le langage à utiliser pour manipuler les paramètres et les formules, reportez-vous à la section [Utilisation des fonctions de règles de connaissance](#) (documentation du programme Infrastructure).

Écriture des règles et des vérifications

Les règles et les vérifications sont des instructions à **plusieurs lignes** que vous pouvez créer, soit en tapant directement les commandes requises dans la zone d'édition, soit en sélectionnant les éléments à partir de la liste du dictionnaire de l'éditeur. La syntaxe à utiliser est décrite ci-après. Les fonctions mathématiques, les fonctions trigonométriques et les fonctions de chaînes sont les mêmes que pour les formules.

[Constantes, unités et opérateurs](#)

[Instructions conditionnelles](#)

[Fonctions d'affichage d'un message ou de démarrage d'une macro](#)

[Fonctions de mesure dans les règles](#)

Constantes, commentaires, unités et opérateurs

Constantes

Lors de la programmation des règles et des vérifications, CATIA peut reconnaître un certain nombre de constantes. Vous pouvez utiliser ces constantes dans une relation à la place des valeurs proprement dites.

- `faux` - une des deux valeurs admises pour un opérateur booléen
- `vrai` - une des deux valeurs admises pour un opérateur booléen
- `PI` - **3,14159265358979323846** - Rapport de la circonférence d'un cercle à son diamètre.
- `E` - Base du logarithme népérien. La valeur de la constante `e` est approximativement **2,718282**.

Commentaires

Les caractères de commentaire `/*` et `*/` sont pris en charge.

```
/* Règle créée par
CRE 05/03/99 */
if CorpsPrincipal\Esquisse.1\Rayon.3\Rayon > 45mm
{
    LaunchMacroFromFile("Macro1.CATScript")
}
else
/*
    LaunchMacroFromFile("Macro2.CATScript")
*/
Message("Aucune macro exécutée")
```

Utilisation de variables temporaires

Les variables temporaires peuvent être déclarées par le biais du mot clé **let**. Une variable temporaire n'existe plus, en tant que paramètre, une fois la règle exécutée.

```
/*Règle créée par CRE
08/23/99*/
let x = 5 mm
if CorpsPrincipal\Trou.1\Diamètre > x
{
CorpsPrincipal\Trou.1\Activité = faux
}
```

Les variables temporaires doivent être déclarées au début de la règle, avant toute spécification d'instruction conditionnelle.

Unités

Toutes les unités apparaissent dans le dictionnaire.

1. Faites attention à la cohérence des unités utilisées lors de l'écriture d'une règle ou d'une vérification.
2. Les unités s'expriment avec un soulignement au lieu du signe "/" habituel (par exemple N_m2 au lieu de N/m2).

Opérateurs arithmétiques

- + Opérateur d'addition (permet aussi de concaténer des chaînes)
- Opérateur de soustraction
- * Opérateur de multiplication
- / Opérateur de division
- () Parenthèses (utilisées pour grouper les opérandes dans les expressions)
- = Opérateur d'affectation
- ** Opérateur d'élévation à une puissance

Opérateurs logiques

- et Conjonction logique de deux expressions
- ou Disjonction logique de deux expressions

Opérateurs de comparaison

- <> Différent de
- == Egal à
- >= Supérieur ou égal à
- <= Inférieur ou égal à
- < Inférieur à
- > Supérieur à

Instructions conditionnelles

Règles

if ... else ... else if

Exécute de manière conditionnelle un ensemble d'instructions en fonction de la valeur d'une expression. Vous pouvez utiliser des syntaxes contenant des blocs d'instructions :

```
if  
instructions_condition[else instructions_else]  
ou  
if condition  
    { instructions }  
[else if condition-n  
    [ {  
instructions_else } ] ] . . .  
[else  
    [ {  
instructions_else } ] ]
```


Pour des règles courtes et simples, vous pouvez utiliser le format à une ligne (première syntaxe). Toutefois, le format bloc (deuxième syntaxe) est plus structuré et plus souple que le format à une ligne. En outre, il permet souvent de lire, de gérer et de tester les commandes plus facilement.


Les deux clauses **else** et **else if** sont facultatives. Vous pouvez avoir autant d'instructions **else if** que vous le souhaitez après un bloc **if**, mais aucune ne peut apparaître après la clause **else**. Les blocs d'instructions **if** peuvent être imbriqués, c'est-à-dire contenus les uns dans les autres.


Vérifications


instruction1 => *instruction2* (**if** *instruction1* **then** *instruction2*)

Affiche un message (si le type est Warning ou Information) et devient rouge dans l'arbre des spécifications chaque fois que l'*instruction2* est incorrecte et que l'*instruction1* est satisfaite.

OK => KO 

KO => KO 

KO => OK 

OK => OK 

Fonctions d'affichage d'un message ou de démarrage d'une macro

Ces fonctions ne peuvent être utilisées que dans les règles.

[Message](#)

Affiche un message dans une fenêtre d'informations.

[Question](#)

Affiche un message, attend que l'utilisateur clique sur un bouton et renvoie un opérateur booléen indiquant quel bouton l'utilisateur a actionné.

[LaunchMacroFromFile](#)

Exécute une macro externe sur la base d'une règle.

[LaunchMacroFromDoc](#)

Exécute une macro stockée dans un document sur la base d'une règle.

Fonction Message

Description

Affiche un message dans une fenêtre d'informations. Le message peut comporter une ou plusieurs valeurs de paramètre.

Syntaxe

Message(*Chaîne* [# *Chaîne1* # *Chaîne2* ..., *NomParam1* *NomParam2*, ...])

La fonction **Message** requiert un argument obligatoire et accepte plusieurs arguments facultatifs selon que les valeurs des paramètres doivent apparaître ou non dans le message.

Arguments	Description
<i>Chaîne</i>	Obligatoire. Chaîne à faire apparaître dans la fenêtre d'informations (à entrer entre guillemets).
# <i>Chaîne1</i> , <i>NomParam1</i> ...	Facultatif. Lorsque les valeurs des paramètres doivent apparaître dans le message, les arguments doivent être spécifiés de la manière suivante : <ul style="list-style-type: none">• une chaîne entre guillemets avec le symbole # pour chaque valeur de paramètre à faire apparaître ;• autant d'instructions [, <i>NomParamètre</i>] que de valeurs de paramètre déclarées par un symbole "#" dans le message.

Utilisez le symbole "|" pour insérer un retour chariot dans un message.

Exemple

```
Message("Le rayon externe est : # | Le rayon interne  
est : #",  
CorpsPrincipal\Essquisse.1\Rayon.3\Rayon  
CorpsPrincipal\Trou.1\Diamètre)
```

Fonction Question

Description

Affiche un message dans une boîte de dialogue, attend que l'utilisateur clique sur un bouton et renvoie une valeur indiquant quel bouton l'utilisateur a actionné (vrai pour Oui et faux pour Non).

Syntaxe

Question(*Chaîne* [# *Chaîne1* # *Chaîne2* ..., *NomParam1*, *NomParam2*, ...])

La fonction **Question** requiert un argument obligatoire et accepte plusieurs arguments facultatifs selon que les valeurs des paramètres doivent apparaître ou non dans le message.

Arguments	Description
<i>Chaîne</i>	Obligatoire. Chaîne à faire apparaître dans la boîte de dialogue (à entrer entre guillemets).
# <i>Chaîne1</i> , <i>NomParam1</i> ...	Facultatif. Lorsque les valeurs des paramètres doivent apparaître dans le message, les arguments doivent être spécifiés de la manière suivante : <ul style="list-style-type: none">• une chaîne entre guillemets avec le symbole # pour chaque valeur de paramètre à faire apparaître ;• autant d'instructions [, <i>NomParamètre</i>] que de valeurs de paramètre déclarées par un symbole "#" dans le message.

Utilisez le symbole "|" pour insérer un retour chariot dans le message.

Exemple

```
Booléen2 =  
Question("RayonEsquisse est # | Voulez-vous changer cette  
valeur ?"  
CorpsPrincipal\Esquisse.1\Rayon.3\Rayon )
```


Fonction LaunchMacroFromFile

Description

Exécute un fichier CATScript de macro sur la base d'une règle.

Syntaxe

LaunchMacroFromFile(*NomMacro.CATScript*)

La fonction **LaunchMacroFromFile** possède un argument :

<i>Arguments</i>	<i>Description</i>
<i>NomMacro</i>	Obligatoire. A entrer entre guillemets.

Exemple

```
LaunchMacroFromFile ("Macro1.CATScript")
```

Fonction LaunchMacroFromDoc

Description

Exécute une macro stockée dans un document sur la base d'une règle.
Remarque : Une macro est stockée dans un document lorsque vous n'indiquez aucun fichier externe avant l'enregistrement.

Syntaxe

LaunchMacroFromDoc (*NomMacro*)

La fonction **LaunchMacroFromDoc** requiert un argument :

Arguments	Description
<i>NomMacro</i>	Obligatoire. A entrer entre guillemets.

Exemple

```
LaunchMacroFromDoc ("Macro1")
```

Fonctions de mesure dans les règles

Cette section décrit les fonctions que vous pouvez utiliser pour extraire des mesures dans des règles. Les fonctions détaillées dans [Fonctions de mesure dans les formules](#) peuvent également être utilisées dans les règles.

coord	<i>point.coord</i> (x: out <i>Longueur</i> , y: out <i>Longueur</i> , z: out <i>Longueur</i>): TypeVide Renvoie les coordonnées d'un point. Les arguments sont des arguments de sortie. Vous devez déclarer les arguments en tant que paramètres de type <i>Longueur</i> .
centerofgravity	<i>body.centerofgravity</i> (x: out <i>Longueur</i> , y: out <i>Longueur</i> , z: out <i>Longueur</i>): TypeVide Renvoie les coordonnées du centre de gravité d'un corps.

Écriture d'un comportement

Un comportement se caractérise par une série d'instructions VB Script associées à une entité. Ces instructions s'exécutent lorsque l'entité concernée répond à un événement donné. Actuellement, un comportement ne peut être associé qu'aux entités manipulées par glisser/déposer. Les méthodes pouvant être appliquées à l'objet d'événement lors de l'écriture d'un comportement sont décrites ci-après.

GetDroppedFeature

Renvoie l'entité déposée.

GetTargetOfType

Renvoie l'entité sur laquelle une autre entité a été déposée. Indiquez un type d'entité comme argument.

GetX, GetY, GetZ

Renvoie les coordonnées de l'emplacement de la copie.

HasCoordinates

Détermine si la copie est associée à des coordonnées.

Travaux pratiques

Un roulement se définit au travers de paramètres tels que ses cotes principales, ses caractéristiques de charge, ses vitesses limites et sa masse. Ce sont les valeurs de ces paramètres qui déterminent la catégorie à laquelle le roulement appartient. Dans un catalogue, un roulement est identifié par une désignation. Les différents types de roulement sont décrits dans des tableaux qui indiquent les valeurs des paramètres et les désignations.

Nous avons choisi l'exemple du roulement parce que les tableaux de spécification que vous trouvez dans les catalogues des fabricants et revendeurs illustrent assez bien les principes des tables de paramétrage. En outre, le roulement lui-même est un bon exemple pour expliquer comment les composants d'une pièce mécanique peuvent être assortis de contraintes par le biais de relations.

Dans l'exemple ci-dessous, le document de départ s'inspire d'un roulement à billes à rainures profondes. Ce document contient déjà plusieurs relations de règles de connaissance ; il existe des relations supplémentaires pour contrôler la conception du document.

[Préambule](#)

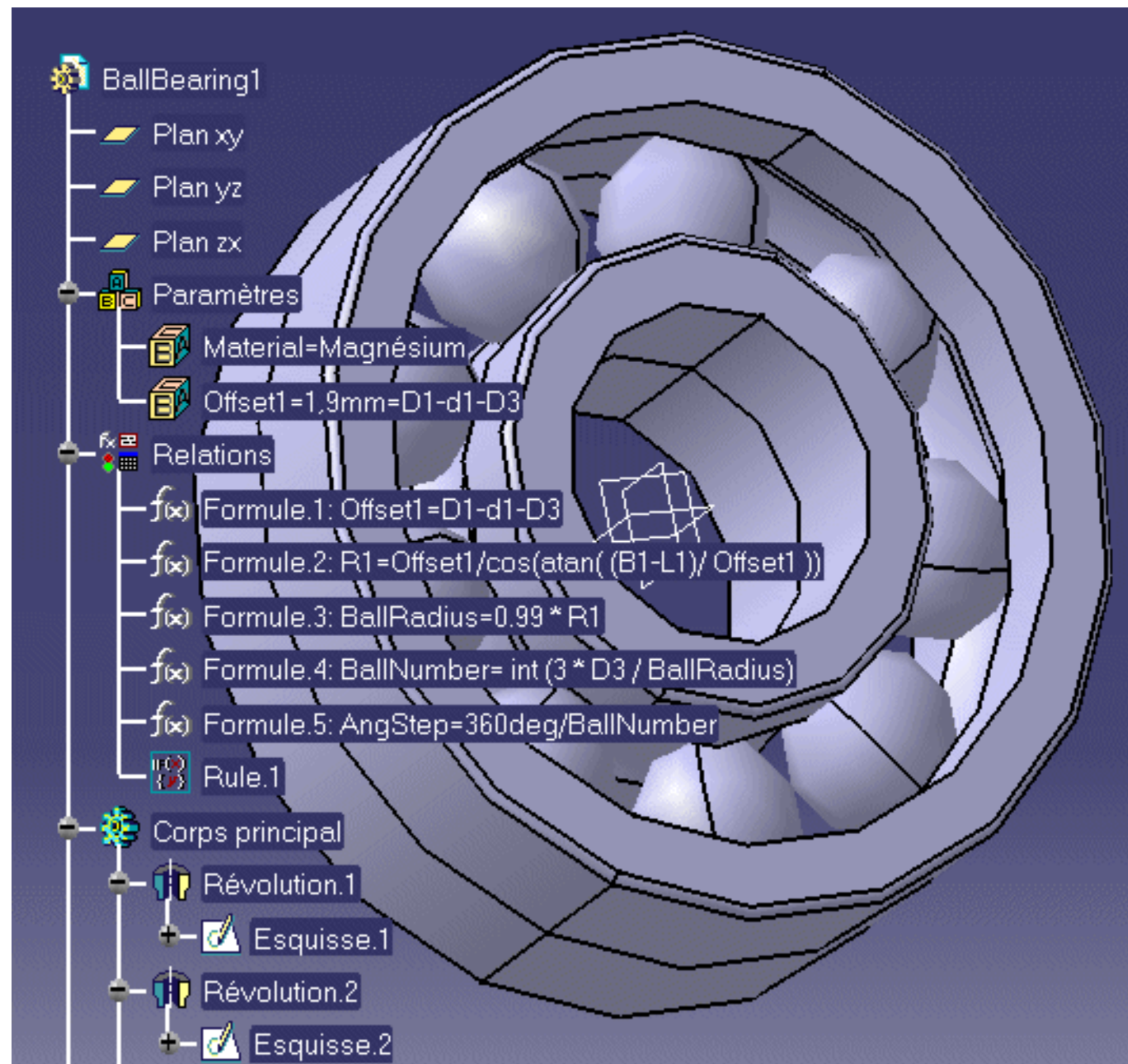
[Etapes de la procédure](#)

Préambule

Cette section décrit les données nécessaires au bon déroulement du scénario. Elles sont toutes fournies avec le programme Knowledge Advisor mais peuvent être recrées à partir des informations données ci-dessous :

Document de départ

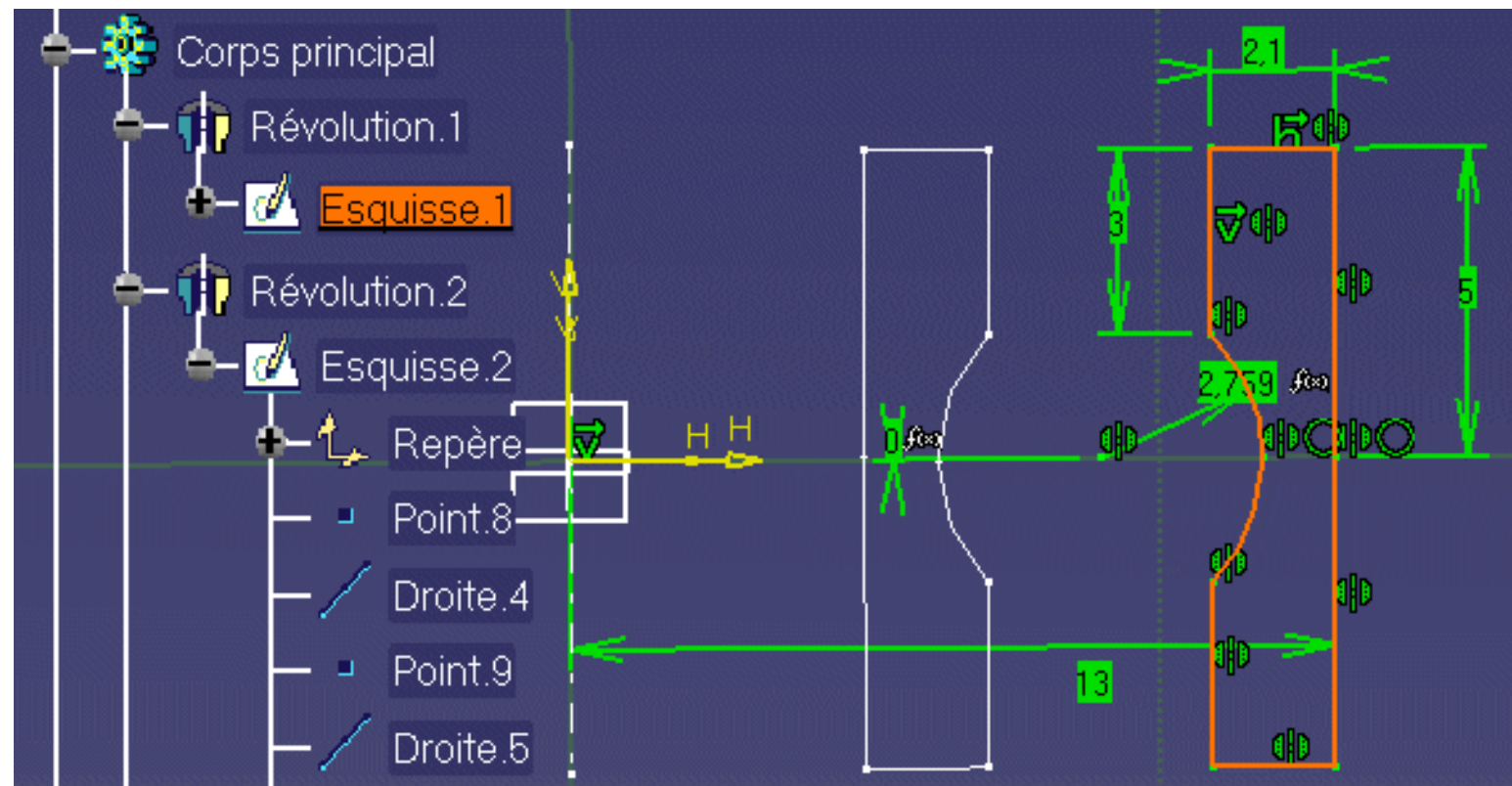
Le document de départ est [KwrBallBearing1.CATPart](#).



Les anneaux du roulement sont des axes coaxiaux créés à partir des composants Esquisse.1 et Esquisse.2. Les billes sont des axes créés à partir de l'entité Esquisse.3.

Anneau externe

L'anneau externe est un axe créé par rotation de l'entité Esquisse.1 mise en évidence dans la figure ci-dessous, autour d'un axe coaxial à V. Notez que vous devez créer cet axe en tant qu'*élément de construction*, faute de quoi CATIA ne vous permettra pas de créer l'axe. Les parties inférieure et supérieure de l'esquisse sont symétriques par rapport à l'axe H.

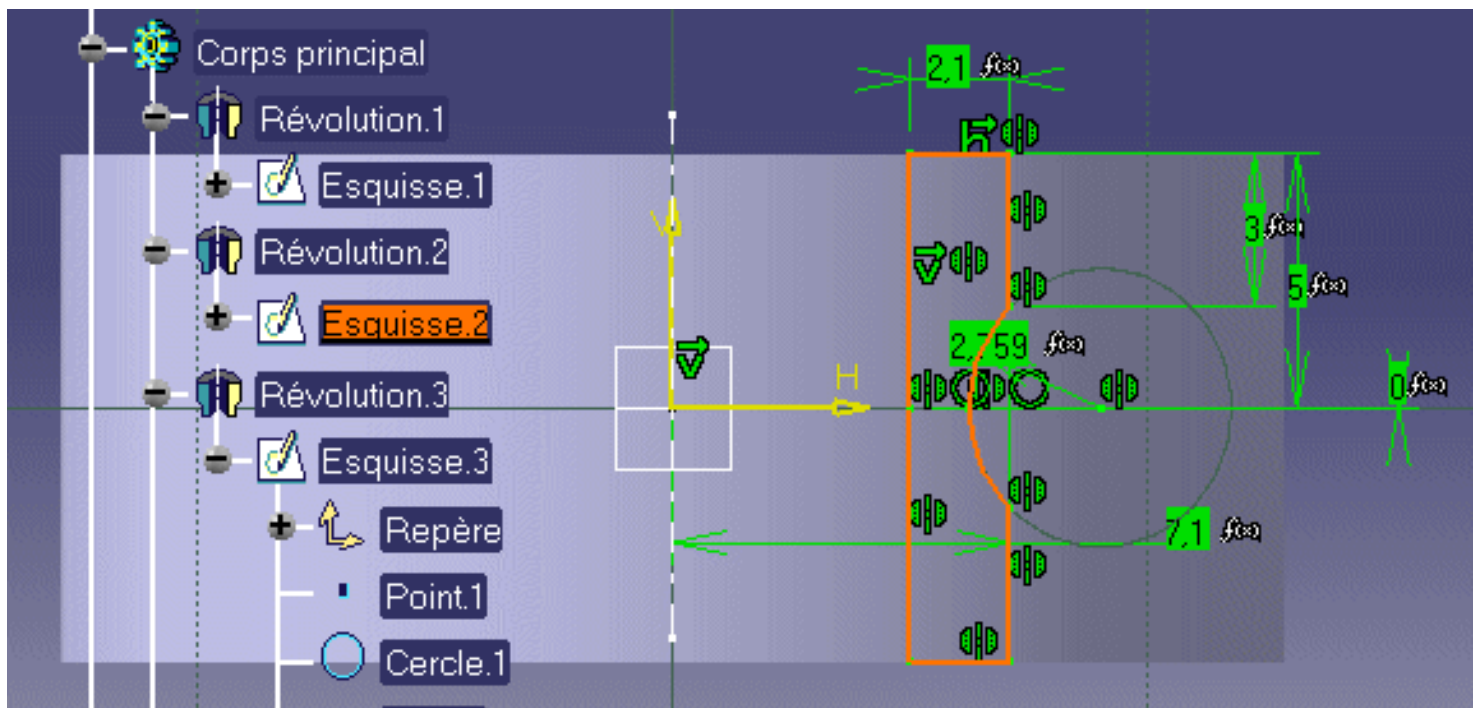


Les contraintes de cette esquisse sont les suivantes :

- d1 2,1 mm *largeur de l'anneau*
- L1 3 mm *mi-hauteur de la surface interne non creusée*
- B1 5 mm *mi-hauteur de la surface extérieure*
- R1 2,759 mm *rayon des rainures*
- b1 0 mm *ordonnée du centre des rainures*
- D1 13 mm *diamètre externe*

Anneau interne

L'anneau interne est un axe créé par rotation de l'entité Esquisse.2 mise en évidence dans la figure ci-dessous, autour d'un axe coaxial à V.

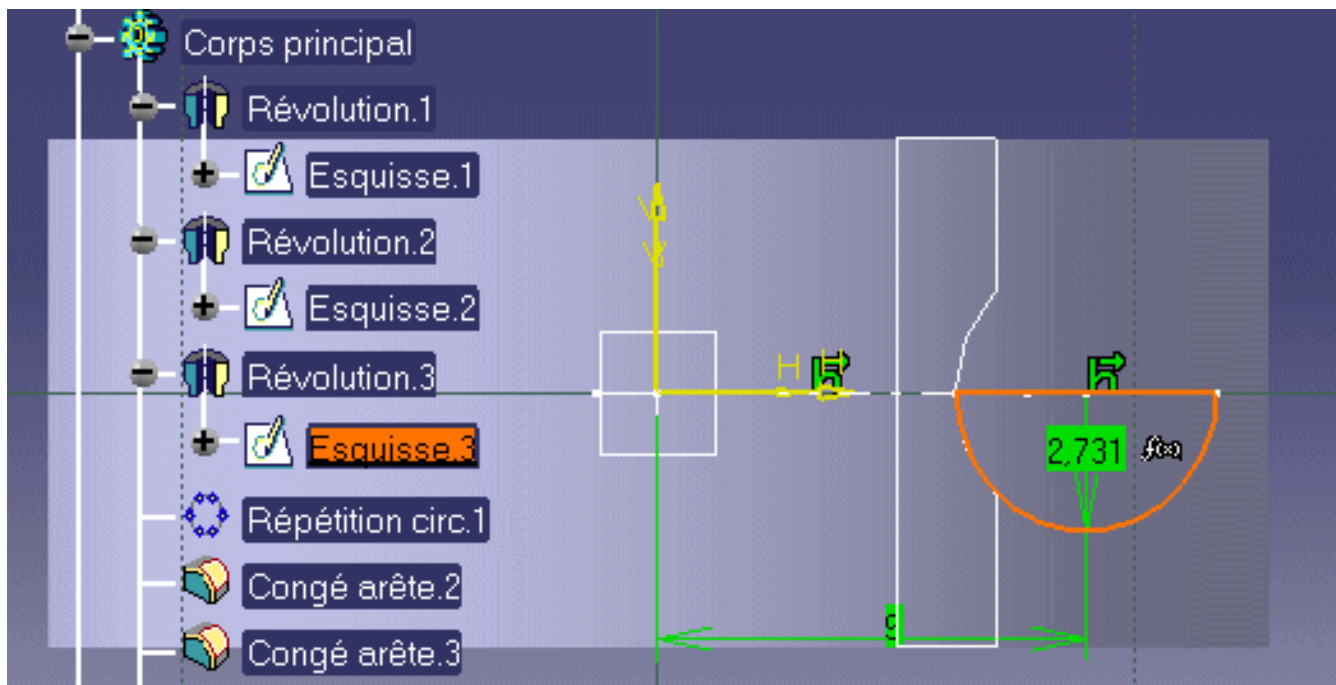


Les contraintes de cette esquisse sont les suivantes :

d2	2,1 mm	<i>largeur de l'anneau</i>
L2	3 mm	<i>mi-hauteur de la surface interne non creusée</i>
B2	5 mm	<i>mi-hauteur de la surface extérieure</i>
R2	2,759 mm	<i>rayon des rainures</i>
b2	0 mm	<i>ordonnée du centre des rainures</i>
D2	7,1 mm	<i>diamètre interne</i>

Billes

Une bille est un axe créé par rotation d'un demi-cercle (esquisse.3) autour de l'axe H. Vous devez fermer le cercle avant d'effectuer la rotation.



Les paramètres du modèle circulaire créé pour générer un ensemble de billes sont assortis des formules de contrainte suivantes :

- $BallNumber = \text{int}(3 * D3 / BallRadius)$
- $AngStep = 3.6\text{deg} / BallNumber$

D3 étant l'abscisse du centre de la bille.

Fichier d'importation

Dans l'exemple, vous devez importer le fichier texte fourni comme modèle [KwrBallBearingImport.txt](#).

```

Temperature      100Kdeg           Maximum temperature
allowed
Pressure         190N_m2           Maximum pressure allowed
LubricantVolume  0mm3           L1*D3*B1*0.005   required lubricant volume
  
```

Si vous modifiez ce fichier, prêtez attention au format des colonnes et utilisez la touche de tabulation pour passer d'une colonne à une autre.

Tableau Excel gérant la conception du roulement

Vous devez télécharger le tableau Excel [KwrBearingDesignTable.xls](#) dans votre environnement.

Macro CATScript

La macro [KwrBearing.CATScript](#) crée uniquement une extrusion circulaire. Vous pouvez vous-même enregistrer cette macro dans l'atelier Part Design ou utiliser celle fournie avec les modèles KnowledgeAdvisor.

Lorsque vous créez la règle **Règle2** dans votre propre environnement, remplacez le chemin d'accès donné comme argument de la fonction LaunchMacroFromFile par celui du répertoire contenant la macro.




Etapes de la procédure

Contrôle de la conception d'un roulement à l'aide d'une table de paramétrage



Une table de paramétrage est créée sur la base d'un fichier existant. Les données contenues dans ce fichier sont semblables à celles qui caractérisent un roulement dans un catalogue. La table de paramétrage créée définit un certain nombre de configurations. L'application d'une nouvelle configuration entraîne la modification du roulement.



1. Ouvrez le document [KwrBallBearing1.CATPart](#).
2. Cliquez sur l'icône  Table de paramétrage dans la barre d'outils standard.
3. Cochez l'option Créer une table de paramétrage à partir d'un fichier existant. Cliquez sur OK.
4. Sélectionnez le fichier [KwrBearingDesignTable.xls](#) et faites une association automatique entre les colonnes de la table de paramétrage et les paramètres du document (c'est-à-dire cliquez sur OUI dans la boîte de dialogue "Associations automatiques ?").
5. Dans la boîte de dialogue "Table de paramétrage", sélectionnez la configuration 3 et cliquez sur Appliquer.
Le roulement à billes a changé. Il est à présent en bronze et contient 21 billes. Vous pouvez remarquer la différence lorsque vous regardez la zone géométrique. La largeur du roulement a également été modifiée. Cliquez sur OK pour quitter la boîte de dialogue Table de paramétrage.
Gardez le document ouvert et passez à la tâche suivante.




Création d'une vérification



CATIA crée une vérification combinée à l'aide de la syntaxe =>. Cette vérification affiche un message chaque fois qu'elle échoue.



1. Ouvrez l'atelier Knowledge Advisor.
2. Cliquez sur l'icône , puis sur OK dans la première boîte de dialogue "Editeur de vérifications". L'éditeur de vérifications s'affiche.
3. Dans l'éditeur de vérifications, sélectionnez le type Warning et entrez la chaîne "Le nombre de billes est trop faible" dans la zone de message. Entrez ensuite la relation
`D3 >= 6mm => BallNumber > 6`
dans la zone d'édition.
4. Cliquez sur OK pour confirmer la création de la vérification et quitter l'éditeur. A ce stade, aucun message ne s'affiche. La vérification ainsi qu'une icône verte sont ajoutées à l'arbre des spécifications. Pour la configuration 3 de la table de paramétrage, le statut des relations de la vérification est le suivant :

OK => OK



5. Dans l'arbre des spécifications, double-cliquez sur la table de paramétrage et sélectionnez la configuration 1. Cliquez sur OK. Le message "Le nombre de billes est trop faible" s'affiche. Pour la configuration 1 de la table de paramétrage, le statut des relations de la vérification est le suivant :

OK => KO



Gardez le document ouvert et passez à la tâche suivante.




Création d'un paramètre à plusieurs valeurs



Un paramètre à plusieurs valeurs est créé. En fonction de la valeur de ce paramètre, une règle créée lors de la tâche suivante affiche un message ou exécute une macro.





1. Cliquez sur l'icône .
2. Dans la boîte de dialogue "Formules", sélectionnez Chaîne dans la liste Créer paramètre de type. Sélectionnez Valeurs multiples dans la liste "Par". **Puis**, cliquez sur Créer paramètre de type.
3. Dans la boîte de dialogue "Liste de valeurs de Chaîne", entrez une à une les valeurs de l'étape 1, 2 et 3. Cliquez sur OK.
4. Dans la boîte de dialogue Editer le nom ou la valeur du paramètre sélectionné, remplacez la chaîne String.1 par Etat, puis cliquez sur OK. Le paramètre Etat est ajouté à l'arbre des spécifications.



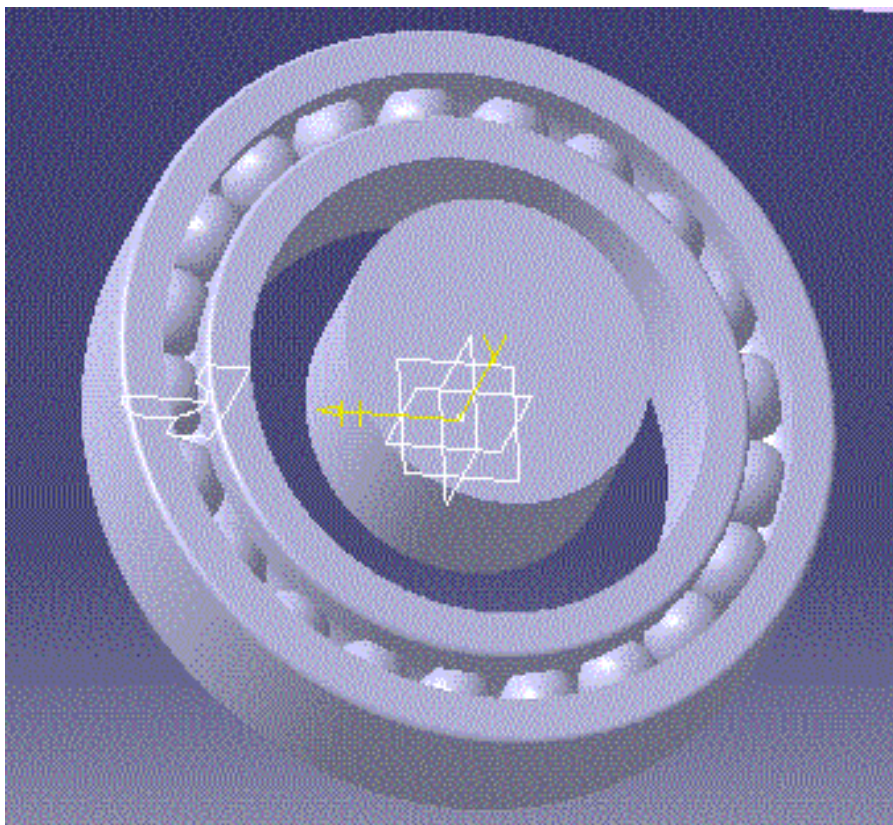
Crée une règle qui affiche un message vous demandant d'importer un fichier ou qui exécute une macro.



1. Dans l'arbre des spécifications, cliquez deux fois sur l'entité de la table de paramétrage et sélectionnez la configuration 3 dans la table qui s'affiche. Le roulement contient à nouveau 21 billes.
2. Ouvrez l'atelier Knowledge Advisor.
3. Cliquez sur l'icône .
4. Entrez la chaîne **Règle.2** dans le champ Nom de la première boîte de dialogue. Cliquez sur OK.
5. Insérez le code ci-dessous dans la zone d'édition des règles avec un copier-coller (modifiez le chemin de la macro) :

```
if Etat == "step2"  
Message("Importez le fichier texte KwrBallBearingImport")  
else if Etat == "step3"  
LaunchMacroFromFile("e:/tmp/KwrBearing.CATScript")
```
6. Cliquez sur OK pour ajouter la règle au document et exécutez-la.
7. Cliquez sur l'icône . Dans la boîte de dialogue Formules, sélectionnez le paramètre Etat et remplacez la valeur step1 par step2. Cliquez sur OK. Un message vous demande d'importer le fichier texte [KwrBallBearingImport](#).
8. Cliquez sur Importer et sélectionnez le fichier [KwrBallBearingImport.txt](#). Trois paramètres viennent s'ajouter au document. Cliquez sur OK dans la boîte de dialogue contenant les paramètres et les formules à importer.
9. Sélectionnez le paramètre Etat et remplacez la valeur step2 par step3. Cliquez sur OK. Le fichier [KwrBearing.CATScript](#) s'exécute et crée une

extrusion circulaire.



Glossaire

.CATScript Extension d'un fichier de macro-instructions généré par CATIA Version 5. Vous pouvez indiquer un fichier de macro-instructions comme argument de la fonction LaunchMacroFromFile, qui peut être invoquée dans les règles et les vérifications.

.txt

Extension d'un fichier composé de caractères texte. Vous pouvez utiliser ce format de fichier d'importation pour importer des paramètres et des formules.

.xls

Extension des fichiers Excel. Vous pouvez utiliser ce format de fichier d'importation pour importer des paramètres et des formules sous Windows.

Activité

Propriété qui définit si une relation est appliquée à un document ou non. La valeur de ce paramètre est soit `vrai` soit `faux`. Elle apparaît sous la forme d'une icône dans l'arbre des spécifications et peut être lue dans la liste des paramètres du document.

Assistant

Programme d'aide qui guide l'utilisateur à travers une tâche complexe, à l'intérieur d'une application. L'assistant des formules aide l'utilisateur à saisir ses formules en lui permettant de sélectionner des paramètres dans le dictionnaire, dans la zone géométrique ou dans l'arbre des spécifications.

Association

Liaison entre un paramètre d'un document et le paramètre équivalent dans une table de paramétrage externe. Les associations sont créées lorsque les noms des paramètres du document ne correspondent pas exactement à ceux lus dans la table de paramétrage.

Comportement

Série d'opérations exécutées en réponse à un événement intervenant sur une entité.

Configuration

Ligne d'une table de paramétrage. Une configuration est une série cohérente de valeurs de paramètre qui peut être appliquée à un document.

Dictionnaire

Ensemble des paramètres, opérateurs, mots clés, fonctions et autres composants appartenant au langage utilisé pour écrire les formules, les règles et les vérifications. L'éditeur de formules, l'éditeur de règles et l'éditeur de vérifications permettent un affichage interactif du dictionnaire.

Formule

Relation imposant une contrainte à un paramètre. La relation de la formule est une instruction composée d'une seule ligne. La première partie de la relation désigne le

paramètre sujet de la contrainte tandis que la deuxième partie contient une relation utilisant d'autres paramètres comme variables. Une formule est une entité. Dans l'arbre des spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée ou désactivée. Comme n'importe quelle entité, une formule peut être manipulée à partir de son menu contextuel.

Inspecteur de Gestion de Connaissance

Outil d'analyse permettant de comprendre comment la modification d'une propriété du document (qu'il s'agisse du matériau, de la pression ou d'une cote) modifie le fonctionnement ou la conception du produit sur lequel l'utilisateur travaille. L'Inspecteur de Gestion de Connaissance propose deux options :

- L'option Analyse des antécédents permet de déterminer comment modifier le produit pour obtenir un résultat donné.
- L'option "Analyse des dépendants" permet d'examiner l'interaction existant entre les paramètres et avec les règles qui composent les spécifications du produit.

Paramètre

Entité définissant une propriété d'un document.

Paramètre de type grandeur

Paramètre dont la valeur se définit par une quantité exprimée dans une unité spécifique. Les paramètres Longueur, Angle, Durée sont des paramètres de type grandeur. Les paramètres Booléen, Réel, Chaîne et Entier n'en sont pas.

Plusieurs des définitions contenues dans ce glossaire ne sont valables que dans le contexte de CATIA.

Règle

Série d'instructions, généralement basées sur des instructions conditionnelles, qui permet de contrôler les relations entre les paramètres. De plus, selon le contexte défini par les instructions de la règle, CATIA peut exécuter des macros ou afficher des messages. Une règle est une entité. Dans l'arbre des spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée ou désactivée. Comme n'importe quelle entité, une règle peut être manipulée à partir de son menu contextuel.

Règles de connaissance

Ensemble des programmes logiciels servant à la création et à la manipulation d'informations basées sur les connaissances. Les informations basées sur les connaissances comprennent les règles et d'autres types de relation, qui permettent aux concepteurs de garder trace des méthodes et des procédures mises en oeuvre et de les réutiliser ensuite pour d'autres travaux de conception.

Relation

Entité de règles de connaissance qui, selon les conditions :

- Affiche un message ;
- Définit des valeurs de paramètre ;
- Exécute une macro.

Les relations de règles de connaissance sont les formules, les vérifications, les règles et les tables de paramétrage.

Table de paramétrage

Tableau contenant des valeurs pouvant être appliquées à un document pour gérer les valeurs de ses paramètres. Ce tableau peut être créé à partir des paramètres du document ou d'un fichier externe. Une table de paramétrage est une entité. Dans l'arbre des spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée ou désactivée. Comme n'importe quelle entité, une table de paramétrage peut être manipulée à partir de son menu contextuel.

Vérification

Série d'instructions permettant de savoir si certaines conditions sont remplies ou non. Une vérification ne modifie pas le document auquel elle s'applique. Une vérification est une entité. Dans l'arbre des spécifications du document, elle apparaît sous la forme d'une relation qui peut être activée et désactivée. Comme n'importe quelle entité, une vérification peut être manipulée à partir de son menu contextuel.

| (opérateur)

Répartit sur plusieurs lignes un message initialement contenu sur une seule. Cette option ne peut être utilisée qu'avec la fonction Message lors de la programmation de règles et de vérifications.

·
A
C
D
F
I
P
R

T
V
I

Conventions

Certaines conventions utilisées dans la documentation CATIA, ENOVIA & DELMIA vous aideront à reconnaître et à comprendre un certain nombre de spécifications et de concepts importants. Les conventions typographiques suivantes sont utilisées :

- Les titres des documents CATIA, ENOVIA & DELMIA apparaissent *en italique* dans le texte.
- Le texte qui apparaît en courier, comme `Fichier` -> `Nouveau`, identifie les commandes à utiliser.

L'utilisation de la souris diffère selon le type d'opération que vous devez effectuer.

Utilisez ce bouton, quand vous lisez



Sélectionner (un menu, une commande, une géométrie dans une zone graphique, etc.)

Cliquer (sur une icône, un bouton dans une boîte de dialogue, un taquet, etc.)

Double-cliquer

Cliquer en maintenant la touche Maj enfoncée

Cliquer en maintenant la touche Ctrl enfoncée

Cocher (une case)

Faire glisser la souris

Faire glisser (une icône sur un objet, un objet sur un autre)



Faire glisser la souris

Déplacer



Cliquer à l'aide du bouton droit de la souris (pour sélectionner un menu contextuel)

Les conventions graphiques sont les suivantes :



indique le temps nécessaire pour exécuter une tâche.



indique la cible d'une tâche.



indique les conditions prérequis.



indique le scénario d'une tâche.



indique des conseils.



indique un avertissement.



indique des informations.



indique la fin d'une tâche.



indique des fonctionnalités nouvelles ou améliorées dans cette version.

Les améliorations peuvent également être indiquées par une marge colorée en bleu en face des paragraphes correspondants.

Remarques

CATIA est une marque de DASSAULT SYSTEMES S.A. en France et dans certains pays.

Les termes qui suivent sont des marques d'autres sociétés :

Java	Sun Microsystems Computer Company
OLE, VBScript for Windows NT, Visual Basic	Microsoft Corporation
IMSpot	Intelligent Manufacturing Software, Inc.

D'autres sociétés sont propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans ce document.

Certaines parties de ce produit contiennent des éléments protégés par des droits d'auteur appartenant aux entités suivantes :

Copyright © Dassault Systemes
Copyright © Dassault Systemes of America
Copyright © D-Cubed Ltd., 1997-2000
Copyright © ITI 1997-2000
Copyright © Summit Software, 1992-1996
Copyright © Cenit 1997-2000
Copyright © Mental Images GmbH & Co KG, Berlin/Germany 1986-2000
Copyright © DISTRIM2 Lda, 2000
Copyright © Institut National de Recherche en Informatique et en Automatique (INRIA)
Copyright © Augrin Software
Copyright © Rainbow Technologies Inc.
Copyright © Compaq Computer Corporation
Copyright © Boeing Company
Copyright © IONA Technologies PLC
Copyright © Intelligent Manufacturing Software, Inc., 2000
Copyright © Smart Solutions Limited
Copyright © Xerox Engineering Systems
Copyright © Bitstream Inc.
Copyright © IBM Corp.
Copyright © Silicon Graphics Inc.
Copyright © Installshield Software Corp., 1990-2000
Copyright © Microsoft Corporation
Copyright © Spatial Technology Inc.

Copyright © 2000, Dassault Systèmes. Tous droits réservés.